

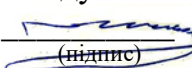
**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет електроніки**  
(повна назва інституту/факультету)

**Кафедра акустичних та мультимедійних електронних систем**  
(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри

 **С.А. Найда**  
(підпис) (ініціали, прізвище)

“ 01 ” червня 2020 р.

**Дипломна робота**

на здобуття ступеня бакалавра

зі спеціальності 171 Електроніка (Електронні системи мультимедія та засоби Інтернету речей)  
(код і назва)

на тему: «Дослідження особливостей створення віртуальних ефектів для сучасного кінематографа»

Виконав: студент III курсу, групи ДВ-п71  
(шифр групи)

Пересецький Вадим Радиславович

(прізвище, ім'я, по батькові)



(підпис)

Керівник

професор, д.т.н., професор Власюк Г. Г.  
(посада, науковий ступінь, вчене звання, прізвище та ініціали)



(підпис)

Консультант

(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали)

(підпис)

Рецензент доцент кафедри ЕПС, к.т.н., доцент Михайлов С.Р.


(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище, ініціали)



(підпис)

Засвідчую, що у цьому дипломному проєкті  
немає запозичень з праць інших авторів без  
відповідних посилань.

Студент

  
(підпис)

Київ – 2020 року

**Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»**

Факультет Електроніки

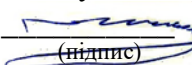
Кафедра акустичних та мультимедійних електронних систем

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 171 Електроніка (Електронні системи мультимедія та засоби Інтернету речей)

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

 С.А. Найда  
(підпис) (ініціали, прізвище)

“01” червня 2020 р.

**ЗАВДАННЯ  
на дипломну роботу студенту**

Пересецькому Вадим Радиславовичу  
(прізвище, ім'я, по батькові)

1 Тема роботи: «Дослідження особливостей створення віртуальних ефектів для сучасного кінематографа»

керівник роботи Власюк Ганна Григорівна, д.т.н., професор  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «25» травня 2020 р. № 1198-с

2 Термін подання студентом роботи 01 червня 2020 р.

3 Вихідні дані до роботи: 1) основний програмний продукт для аналізу – The Foundry NukeX non-commercial; 2) тривалість секвенції з кінофільму для дослідження – до 5 сек; 3) кількість віртуальних камер для налаштування 3D сцени – 1; 4) частота кінокадрів – 24 кадри в сек; /  
/

4 Зміст роботи: 1) Визначити підходи та алгоритми роботи з віртуальними ефектами та композитинг в NukeX; 2) Сформулювати основні напрямки, за якими доцільно проводити налаштування Nuke для 3D сцени; 3) Розглянути особливості переміщення камери в 3D сцені NukeX; 4) Розробити алгоритм

переміщення віртуальних 3D-об'єктів в часі; 5) Проаналізувати основні функції та можливості нодів, які відповідають за рух в сцені, суміщення 3D об'єктів в просторі. 6) Розробити алгоритм створення ефекту вибуху в програмі Houdini.





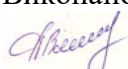
5 Перелік ілюстративного матеріалу: комплект презентації за матеріалами проведеного дослідження.

#### 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання 11 березня 2020 р.

#### Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Написання першого розділу	24.03.2020	Виконано 
2	Написання другого розділу	20.04.2020	Виконано 
3	Написання третього розділу	10.05.2020	Виконано 
4	Підготовка матеріалів до друку та оформлення пояснювальної записки	31.05.2020	Виконано 
5	Підготовка та оформлення презентації для доповіді	2.06.2020	Виконано 

Студент



(підпис)

В.П. Пересецький

(ініціали, прізвище)

Керівник роботи



(підпис)

Г.Г. Власюк

(ініціали, прізвище)

УДК 681.3.06

## РЕФЕРАТ

Дипломна робота: 64 с., 49 рис., 10+ джерел.

ВІРТУАЛЬНІ ЕФЕКТИ, ЗОБРАЖЕННЯ, СЕКВЕНЦІЯ, ФОРМАТ, ТЕХНОЛОГІЯ, 3D, КАМЕРА, РОТОСКОП, КОМПОЗИТИНГ, НОДА, СХЕМА.

**Об'єкт дослідження** – набір віртуальних ефектів та робота з ними, що є основою при створенні сучасного кінофільму.

**Метою роботи** є розроблення послідовності дій, яка дозволяє робити суміщення віртуальних ефектів та проводити налаштування з різних спеціальних видів цифрового композитингу.

**Методом дослідження** є аналіз літератури, досліджень і розробка алгоритму створення віртуальних ефектів у сучасному кінематографі.

У результаті виконання дипломної роботи було визначено особливості створення віртуальних ефектів у сучасному кінематографі. Зокрема, на реальному практичному матеріалі створено три дерева нодів в спеціальній програмі NukeX, які дозволяють робити операції з 3D-об'єктами та розроблено алгоритм створення ефекту вибуху у програмі Houdini.

Галузь застосування: кіно виробництво, цифровий композитинг.

## **ABSTRACT**

The object of research is a set of virtual effects and work with them, which is the basis for creating a modern film.

The aim of the work is to develop a sequence of actions that allows you to combine virtual effects and make settings from different special types of digital compositing.

The research method is the analysis of literature, research and development of an algorithm for creating virtual effects in modern cinema.

As a result of the thesis, the features of creating virtual effects in modern cinema were identified. In particular, three node trees in a special NukeX program were created on real practical material, which allow to perform operations with 3D objects, and an algorithm for creating an explosion effect in the Houdini program was developed.

Field of application: film production, digital compositing.

## ЗМІСТ

Abstract.....	5
ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	7
ВСТУП.....	8
1 ВІРТУАЛЬНІ ЕФЕКТИ. ЗАСТОСУВАННЯ В КІНЕМАТОГРАФІ.....	9
1.1 Історія віртуальних ефектів в кіно.....	9
1.2 Зміна частоти кінозйомки за допомогою спецефектів.....	16
Висновки до розділу 1.....	18
2 РОБОТА З ЕФЕКТАМИ ТА КОМОЗИТИНГ В NUKE.....	19
2.1 Налаштування Nuke для 3D сцени.....	19
2.2 Переміщення камери в 3D сцені.....	21
2.3 Створення 3D об'єкта використовуючи пас світового положення.....	23
2.4 Імпорт камери.....	25
2.5 Імпорт системи частинок в 3D ДЕРЕВА NODE .....	29
Висновки до розділу 2.....	32
3 СУМІЩЕННЯ ЕФЕКТІВ У ПРОГРАММІ NUKE.....	33
3.1 Суміщення 3D-об'єктів в просторі.....	33
3.2 Переміщення 3D-об'єктів в часі.....	35
3.3 Переведення 3D об'єктів в 2D пікселі.....	36
3.4 Застосування матеріалів до об'єктів.....	40
3.5 Корекція кольору зображення.....	44
Висновки до розділу 3.....	47
4 РОБОТА З ВІРТУАЛЬНИМ ЕФЕКТОМ ВИБУХУ В ПРОГРАМІ HOUDINI.....	48
4.1 Створення симуляції віртуального ефекту вибуху.....	48
4.2 Налаштування шейдера fireball та Render.....	54
Висновки до розділу 4.....	58
ВИСНОВКИ.....	59
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	61

## **ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ**

CAF	- Content-Aware Fill;
CGI	- Computer generated graphics;
CAM	- Content-Aware Move;
DAG	- Define a node graph;
DCP	- Digital Cinema Packet;
FOV	- Field of view;
FBX	- Flimbox;
G	- Gamma;
OBJ	- Object;
RGB	- Red, Green, Blue;
Src	- Source;
SFX	- Special effects;
VFX	- Visual effects.

## ВСТУП

Дуже швидкий розвиток кінематографії потребує зручних, доступних та зрозумілих інструментів, що дозволяють значно скоротити час на створення кінофільмів зі віртуальними ефектами та спецефектами. Спецефекти різної складності в сучасному кінематографі використовуються практично у всіх фільмах. Екран без них втратив би свою актуальність і вражаючу силу. Ми можемо бачити як на телебаченні так і в кіно, в кожному фільмі або серіалі автори для вирішення своїх проблем використовують комп'ютерні можливості формування видовища, комп'ютерні технології, програми для монтажу та віртуальних ефектів. Ми бачимо неймовірні речі на фоні головних героїв, зіставлення і поєднання незвичайних зображень, також речі, які практично не існують в природі, і багато-багато іншого. У сучасному кіно творці фільмів можуть відтворити все, що вони хочуть за допомогою комп'ютерної графіки, і в зв'язку з цим можна сказати, що сьогодні в кіно немає меж.

З ростом використання віртуальних ефектів у фільмах, телесеріалах і рекламних роликах, стає необхідною висока продуктивність при створенні динамічних зображень. Саме з цих обставин вважаю обрану тему дослідження актуальною.



## **1 ВІРТУАЛЬНІ ЕФЕКТИ. ЗАСТОСУВАННЯ В КІНЕМАТОГРАФІ**

Віртуальні ефекти та спецефекти - технологічний прийом в кінематографі, на телебаченні, в комп'ютерних іграх та програмах, застосовуваний для візуалізації сцен, які не можуть бути зняті звичайним способом (наприклад, як для візуалізації сцени битв космічних кораблів в далекому майбутньому)[1].

Спецефекти також часто застосовуються, коли природна зйомка сцени занадто затратна в порівнянні зі спецефектом (наприклад, зйомка масштабного вибуху). Віртуальні ефекти застосовуються і для поліпшення або модифікації вже попередньо відзнятих відеоматеріалів (наприклад, для накладення погодної карти на фон для телеведучого на телеканалх у прямому ефірі, який розповідає про прогноз погоди).

### **1.1 Історія віртуальних ефектів в кіно**

Проникнення цифрових технологій в кінематограф найбільш інтенсивно почалося з етапу обробки зображення, етапу, якого, в кінематографі у доцифрову епоху практично і не існувало. Зараз же стало мало не правилом хорошого тону піддавати зображення всякого роду маніпуляцій. Цифрові фотоапарати, в свою чергу, лише порівняно недавно стали відвойовувати собі позиції на знімальному майданчику. Така апаратура має безліч переваг, вона значно компактніше, менш вимоглива до освітлення, зручна у використанні. Але при цьому до недавнього часу вона все ж не могла забезпечити необхідну для кіно якість зйомки. Лише в останні роки з'явилися камери відповідного рівня. Одну з них запропонувала компанія Sony.

Першим в повнометражному кіно з її можливостями ознайомився французький режисер Пітоф. По ходу роботи над фільмом «Відок» він пробував використовувати Camcorder HDW-F900 24p при слабкому освітленні, в диму, для створення того чи іншого настрою зйомки. Наступним став Джордж Лукас.

Вилучену Галактику в картині «Зоряні війни: Епізод 2», де мешкали кошмарні потвори і відбувалися епічні битви, збудували на Fox Studios в Сіднеї. Незважаючи на те що в день змінювалося до тридцяти шести різних декорацій, а працювати доводилося і в дощ, і в спеку, як в пустелі, камера 24p не створювала ніяких складнощів. На цій же камері зупинив свій вибір і Олександр Сокуров у фільмі «Російський ковчег», де стояло завдання зняти 90 хвилин екранної дії одним планом.

Стівен Спілберг дотримується прямо протилежної точки зору. «Я відчуваю дуже сильну ностальгію по колишньому кіно і просто не можу знімати цифровою камерою. Напевно, я останній в Голлівуді, хто монтує фільми по-старому, на плівці. Всі великі в історії кіно картини склалися за монтажним столом, на плівці, і я чіпляюся за цей рік, що минає процес »[1], - сказав він в одному з інтерв'ю. Хоча, як не парадоксально, саме Спілберг йде в авангарді, якщо говорити про використання комп'ютерних технологій на етапі обробки зображення, про що свідчить хоча б його «Парк Юрського періоду».

З великими труднощами цифрові технології прокладають собі шлях в прокаті, хоча і тут можуть запропонувати чимало переваг.

І все ж найбільш відчутний вплив цифрові технології надають саме у вигляді CGI, тобто на етапі обробки зображення. Уявити собі сьогоднішній блокбастер без комп'ютерних ефектів практично неможливо. Взяти хоча б «Піратів Карибського моря», особливо дві останні стрічки. Вони просто рясніють демонстрацією досягнень CGI. У «Піратах» дійсно застосований майже весь арсенал комп'ютерних трюків: поєднання в просторі предметів з різних місць (пейзаж з одного району, корабель з іншого, небо з третього), моделювання «живих» акторів, використання віртуальних ефектів анімації замість залучення дублерів, «створення» явищ природи, анімація великих груп об'єктів і т.п.

Всього в третьому фільмі налічується 2000 ПЛАН зі спецефектами, з яких 752 виконано ILM, однією з провідних студій в області віртуальних ефектів. Решта план-епізоди створені фахівцями декількох студій - CIS Hollywood, Digital Domain, The Orphanage, Asylum VFX, Evil Eye Pictures, Method Studios і Luma

Pictures. У команді ILM, якою керував Джон Нолл, працювали (менше трьох місяців) близько двохсот професіоналів.

Обсяг підготовленого ними матеріалу склав 103 терабайта інформації. Для порівняння згадаємо, що, за оцінкою фахівців, щоб вмістити весь матеріал, який є в Бібліотеці Конгресу - а це двадцять мільйонів томів, - потрібно 20 терабайт. А для того щоб перетворити всі ці терабайти в зорові образи, обчислювальна техніка щодня напружувалася 100 тисяч процесорних годин.

Кульмінація фільму - двадцятихвилинний епізод з гігантським вибухом. «Летючий Голландець» під командуванням Дейві Джонса і «Чорна Перлина» Джека Горобця плывуть назустріч один одному. Намагаючись втриматися на краю обертається воронки в півмилі завширшки, противники відкривають вогонь. Але сила стихії занадто велика, і судна починають занурюватися в воду, обертаючись все швидше і швидше.

Спочатку воронку збиралися створити, наклавши текстуру океанічної води на посудину конічної форми. Саме такий звичайний порядок дій, коли потрібно зобразити поверхню моря. Але вже через кілька хвилин після початку першого ж тестового перегляду стало ясно, що такий «вир» виглядає неприродно, бо дуже примітивно. Довелося вдатися до більш складної системи імітації рідин, в якій ступінь правдоподібності безпосередньо залежить від наявних у розпорядженні обчислювальних ресурсів. Чим більше інформації про окремі «крапельки» можна обробити (їх розмір, положення в просторі, яскравість і т.п.), тим вірогідніше результат. У 2006 році під час зйомок «Посейдона» обчислення доводилося розділяти між декількома одночасно працюючими процесорами. Тепер же, незважаючи на величезне зростання продуктивності техніки, завдання настільки ускладнилися, що потрібна була обчислювальна потужність сорока процесорів.

Але на цьому «малювання» виру не завершилося. Попереду було ще додати бризки, туман над водою, серпанок в повітрі. На щастя, в двохстах п'ятдесяти з трьохсот кадрів основна дія відбувалося на палубі, а воронка лише трохи виднілася на задньому плані, що давало можливість обійтися без докладної опрацювання всіх деталей.

На довершення всього, за задумом режисера, що потрапили у вир кораблі повинні були рухатися з різною швидкістю, тоді як в дійсності рух всередині подібного природного феномена відбувається рівномірно по всій поверхні. Але для CGI порушення законів фізики - справа звична. Вихід був знайдений: одну і ту ж імітацію руху всередині виру провели двічі з різною швидкістю, а потім поєднали обидва фрагмента.

Взагалі, що надається CGI можливість обходити закони фізики безцінна для кінематографії. Взяти хоча б приклад з «Матриці». У сцені погоні на автостраді стало можливим зіштовхнути два рухомих з величезною швидкістю транспортних засоби і змусити їх випробувати руйнівну силу сюрреалістичної інерції. Неймовірні події можна знімати в неймовірних ракурсах. Основи ж цього методу були закладені в першій «Матриці», коли в сцені краху вертольота по фасаду висотного будинку, ніби по водній гладі, пробігали хвилі. В алгоритмічній світі «Матриці» такі відступу від законів фізики є цілком допустимими.

У наступних серіях вони стали ще помітніше. «Набагато цікавіше руйнувати що-небудь в кіно таким чином, як це неможливо зробити в реальному житті, ніж просто показувати, як влаштувати хаос за допомогою інгредієнтів, які легко придбай в місцевому супермаркеті»[1], - говорив Джон Гаета, що відповідав у цих фільмах за створення основної частини віртуальних ефектів.

Не менш вражаючих ефектів можна домогтися і маніпулюючи часом. Так зване «час кулі», що з'явилося в «Матриці», набуло широкого поширення, проклавши собі дорогу навіть в останні роботи Такесі Кітано.

Виглядає це приблизно так: двоє людей стоять на даху. Один з них, одягнений в чорний піджак, з чорною краваткою, пильно дивиться на іншого крізь чорні окуляри. І раптом випускає в нього кілька куль. А далі час сповільнюється, поки кулі летять у напрямку до жертви. Ракурс змінюється, в той час як людина прогинається назад, ухиляючись від куль. Смертоносний свинець проноситься мимо вже з нормальною швидкістю. Людина випрямляється і продовжує бій.

Щоб здійснити подібний показ кінематографічної дії в стилі японського аніме, потрібно здійснити сповільнену зйомку камерою, що рухається з постійною швидкістю. Щоб досягти, здавалося б, неможливого, команда розробників спецефектів розташувала 120 фотокамер «Нікон» уздовж траєкторії, розрахованої на комп'ютері. Камери послідовно робили фотографії дії, потім знімки сканувалися. Комп'ютер генерував і доповнив відсутні моменти між кадрами, і вже після цього сцена була з'єднана з цифровим фоном. Але досліджуючи «Пірати Карибського моря», то протягом всього епізоду з вибором цифрові і реальні персонажі взаємодіють один з одним. Зйомки епізоду проходили в ангарі в Палмдейле, штат Каліфорнія. Актори грали на палубі повнорозмірних макетів «Голландця» і «Перлини». Навколо цієї величезної декорації розмістився «синій екран» висотою 65 футів і площею 650 тисяч квадратних футів, що дозволяв вести зйомку з поворотом на 270 градусів. Завданням ІЛМ було «витягти» акторів і палуби з відзнятого матеріалу. Використання технології «захоплення руху» дозволило перетворити гру акторів і каскадерів в обширні дані для цифрової анімації команди «Голландця». Потім акторів і палуби «трансплантували» на цифрові кораблі з цифровими вітрилами і оснащенням, наповнили кораблі цифровими матросами і пустилися обертатися в вирі.

Як відомо, комп'ютерні технології дозволяють створювати фотореалістичних героїв «з нуля». Таким персонажем в «Піратах» став Дейві Джонс. Якщо при роботі над другою серією основним завданням було гладко і непомітно ввести в дію Дейві Джонса, для чого було потрібно написати й налагодити відповідне програмне забезпечення, то тепер все необхідне вже було в розпорядженні ІЛМ і можна було займатися більш творчими завданнями. В останній стрічці Джонс змагається на мечях, впадає в лють, навіть проливає сльозу. Як і раніше, 95 відсотків часу симуляції йшло на «пожвавлення» щупалець персонажа. На щастя, ця частина роботи була вже настільки налагоджена, що цього разу відповідно до вказівок режисера Гора Вербінські їх можна було змушувати «грати».

Найскладнішим виявилось зовні зовсім непримітний кадр, в якому Дейві Джонс проливає сльозинку. Адже потрібно було зібрати крапельку рідини в складках шкіри капітана, а потім, коли він моргне, зробити так, щоб вона скотилася по щоці. Доводилося враховувати і поверхневий натяг рідини, і складність руху краплі, що скачується по нерівній поверхні шкіри, і що залишає нею вологий слід.

З'явилися у персонажа емоції, переживання відразу ж зробили його більш цікавим, ніж стандартне примітивне чудовисько.

Дейві Джонс - це завжди цифровий персонаж, і створювався він методом так званого «захоплення руху».

Найбільш яскравий приклад використання «захоплення руху» в кіно - це Голлум з трилогії «Володар кілець». Автори поставили перед собою завдання зробити Голлума більш правдоподібним цифровим персонажем, здатним взаємодіяти з партнерами. Це результат синтезу акторської гри і цифрових технологій. Кожен рух, кожен вираз обличчя створювалося актором, а потім засобами комп'ютерної обробки перетворювалося в щось зовсім нелюдське. Це класичний випадок, коли в одному персонажі уживаються екранний герой і лиходій. За допомогою одного лише гриму або одних тільки комп'ютерних технологій не вдалося б відтворити всю складну гаму переданих актором емоцій.

Кожна сцена знімалася як мінімум тричі: з Енді Серкісом, одягненим в білий обтягуючий костюм, за участю інших акторів; другий раз Серкіс говорив за кадром, а партнери на знімальному майданчику зверталися в порожнечу, де потім з'явиться анімований Голлум; і нарешті, Серкіс грав один в студії в спеціальному костюмі, за допомогою якого всі його рухи оцифровувати. Саме цей процес реєстрації рухів актора відомий як «захоплення руху», або motion capture. Дивлячись на Голлума, забуваєш, що це цифровий персонаж, адже Серкіс надав стільки додаткової емоційної та психологічної інформації.

Уже розроблено чимало варіацій технології «захоплення руху». Одна з них, так звана performance capture, широко застосовувалася в стрічці «Полярний експрес» (2004). Це перший CGI-фільм, де «захоплення руху» використовувався

для всіх дійових осіб, завдяки чому анімовані персонажі вийшли на подив «живими», аж до того, що інколи виникало відчуття, що на екрані не цифровий персонаж, а все-таки цілком реальний хлопчик. Крім рухів тіла в даному випадку фіксувалася також і міміка.

Це найбільш явні прояви цифрової обробки зображень та створення віртуальних ефектів зображень у кінематографі. Не менше значення має і непомітне застосування CGI, як, наприклад, в такому фільмі, як мюзикл «Бріолін» Адама Шенкмана, де комп'ютерні технології віртуальних ефектів потрібні були для того, щоб привести міський пейзаж сучасного Балтимора у відповідність з 60-ми роками минулого століття. Не випадково фахівці з комп'ютерних ефектів, що відповідають за виконання «чорної роботи» щодо виправлення всіляких недоліків, високо цінуються творцями фільмів. Так, за те, щоб прибрати обриси нижньої білизни, виразно проступають під костюмом Марлона Брандо в «Острові доктора Моро», продюсери не поскупилися заплатити студії ILM 50 тисяч доларів.

Якщо розглядати кожен випадок використання CGI в кіно ізольовано, то стає очевидним, що за рідкісним винятком аналогічного результату можна було б досягти і традиційними засобами. Адже перші приклади морфинга, або перетворення одного предмета або істоти в щось інше, зустрічаються ще у Мельєса, аналоги фракталів - в «Космічної Одиссеї», коли глядач спостерігає захоплюючі дух абстрактні геометричні зображення в епізоді польоту крізь час, різні маніпуляції з кольором і атмосферою - у Куросави в картині «Я бачив такі сни». Використання CGI істотно полегшує вирішення подібних завдань, а число отриманих таким чином кадрів настільки велике, що постає питання вже про цифрові фільми.

Цифрова обробка і комп'ютерне створення зображень іноді змушують засумніватися в таких базових для кінематографа поняттях, як фотографічний реалізм, індексна природа кіно. Не виключено, що уточнення вимагає саме визначення фільму, який перестав зніматися на плівці, демонструватимуться в кінозалі, не є більш фіксацією передкамерної реальності або взагалі якої б то не було реальності.

Цифрове зображення змінює наше відчуття необхідного зв'язку між камерою і реальністю. Присутність того і іншого вже не є абсолютно необхідним. Тепер «сфотографувати» те, що неможливо побачити, стало набагато простіше. Комп'ютерні технології та ефекти перетворюють зображення в пікселі, які можна з легкістю трансформувати, переробляти, змінювати. Стирається лінія між анімацією (яка створює образи там, де їх взагалі раніше не було) і монтажем (який займається перестановкою фрагментів подій, що сталися перед камерою). Коли художник отримує можливість легко маніпулювати цифровим зображенням або в цілому, або покадрово, фільм перетворюється в серію малюнків. Можливість вручну малювати на оцифрованих знімках - це дуже серйозна зміна в статусі кіно, парадоксальним чином повертає «мистецтво рухомих зображень» до його витоків.

## **1.2 Зміна частоти кінозйомки за допомогою спецефектів**

Стоп-кадр. Використовується в сценах, де потрібно показати раптову появу або зникнення об'єкта або змінити природний інтервал між двома подіями.

Наплив. Кінематограф початку і середини XX століття використовував цей прийом для показу «перетворення» одного об'єкта в інший. На один і той же відрізок кіноплівки на тому ж фоні знімається спочатку один об'єкт з поступовим зменшенням експозиції (закриттям діафрагми об'єктива або - частіше - зменшенням кута розкриття обтюратора), а потім - інший з поступовим збільшенням експозиції. У підсумку на екрані зображення одного об'єкта плавно заміщається іншим[6].

Покадрова зйомка. Метод використовується в основному при роботі з макетами; широко застосовувався в середині XX століття при створенні фільмів про чудовиськ (наприклад, «Кінг-Конг»). Ця ж технологія застосовується для виготовлення лялькових і пластилінових мультфільмів (анімації). Покадрова зйомка дозволяє зробити видимими повільні процеси, невидимі оком: зростання рослин, добовий рух світил і т. п.[6а].



Прискорена кінозйомка або «rapid». Надшвидкісна зйомка процесу, що протікає дуже швидко (наприклад, вибух). Потім плівка прокручується в звичайному темпі, що дозволяє отримати уповільнене високоякісне зображення.

Уповільнена кінозйомка, або «прискорене відтворення». Ефект, зворотний рапиду - зйомка ведеться на меншій швидкості, а потім прокручується в нормальному або прискореному темпі. Незначне прискорення використовується в сценах поєдинків, коли треба зняти стрімкі рухи, які актори просто не встигали б виконувати. Сильне прискорення може використовуватися для створення комічного ефекту або для відображення великих відрізків часу.

Зворотня зйомка. Наприклад, коли потрібно зняти злітаючу людину, простіше зняти падіння, а потім прокрутити плівку в зворотному напрямку.

Спецефекти поділяють на візуальні - це оптичні ефекти і комп'ютерна графіка, механічні та звукові спецефекти.

Режисер фільмів "Чужі", "Титанік" і "Термінатор" - Джеймс Кемерон. На той момент, коли Кемерон придумав історію про робота-убивцю, що бореться проти людей у майбутньому. Анімація перетворення T-1000 проходила через п'ять роздільних стадій: від аморфної краплі рідкого металу через стадію "хромованої людиноподібної істоти" до реалістичної на вигляд людини. Робота над кадрами була розділена між шістьма командами художників і аніматорів. Наприклад, "Група руху людини" відповідала за створення хромованого тулуба T-1000 - передостанній стадії трансформації термінатора.

У стрічці "Гаррі Поттер і кубок вогню" Лиходій Волан-де-Морт відрізняється від своїх "побратимів" відсутністю носа і наявністю лише тонких прорізів. Спочатку передбачалося "видалення" носа за допомогою макіяжу, але результат його накладення був посереднім. Було вирішено використовувати засоби комп'ютерних чарівників, які "видалили" ніс актора. Для цього на обличчя актора Ральфа Файнса завдали маркери, які були контрольними точками для накладення комп'ютерного зображення на його ніс.

## Висновки до розділу 1

Віртуальні ефекти та спецефекти в кінематографі застосовуються досить давно і дуже широко. Комп'ютерні технології у сучасному кінематографі розкривають безліч нових можливостей перед режисерами, операторами, постановниками.

Спецефекти були народжені майже у той же час що і кінематограф. Віртуальні ефекти також часто застосовуються, коли на природну зйомку сцени буде витрачено занадто багато коштів в порівнянні зі спецефектом (наприклад, якщо це дуже масштабний вибух якогось спорудження). Спецефекти вікористовуються і для поліпшення або модифікації вже попередньо відзнятих відеоматеріалів (наприклад, вставка фонового зображення за телеведучим, який розповідає про прогноз погоди).

## 2 РОБОТА З ЕФЕКТАМИ ТА КОМОЗИТИНГ В NUKE

Найпотужніша функція в Nuke це його 3D-модуль. Nuke має майже всю багатофункціональну 3D систему, яка може імпортувати всі камери, робити прості об'єкти, робити редагування і імпортування об'єктів, робити проєкції камер, витягувати різні типи 2D даних з 3D контенту та багато чого іншого. Хоча можливості створювати віртуальні ефекти в Nuke є чудовими для системи композитинга, вони все ще не в повній мірі виконують роботу, щоб замінити повноцінні 3D пакети. Також 3D модуль Nuke використовують, щоб легше виконувати 2D трекінг інформації, спростити ротоскопінг і багато іншого. Nuke також добре працює із зовнішніми 3D-додатками і може обмінюватися даними з ними, використовуючи формати файлів, зазвичай використовуваних в 3D-додатках: Alembic, OBJ і FBX. Розширення .obj для файлів, які мають лише геометрію. Alembic і FBX типи файлів можуть містити практично все, що генерується в 3D сцені[2].

### 2.1 Налаштування Nuke для 3D сцени

Алгоритм створення віртуальної камери в Nuke:

1. Спочатку треба, створити вузол камери з 3D-інструментів.
2. Другий крок, створити вузол сцени з 3D-інструментів.
3. Для цього потрібно обрати CAMERA1 і Scenel і далі вставити вузол ScanlineRender з 3Dінструментів. Вузол ScanlineRender поєднує себе з іншими вузлами камери і сцени в правильних входах (рис 2.1).

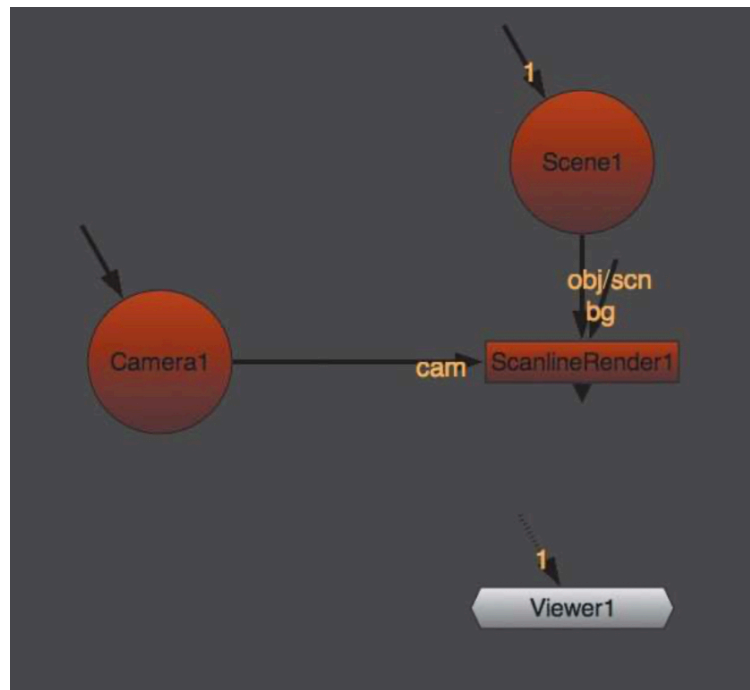


Рисунок 2.1 – Перші кроки налаштування 3D-сцени

4. Обираємо ScanlineRender1 і натискаємо клавішу 1, для перегляду його. На даний момент у вікні перегляду ми нічого не бачимо в режимі 2D зображення. Проте, в 3D-світі в нас вже існує камера. В Nuke також є такі функції як 2D і 3D Viewer. Між двома режимами для перегляду 2D і 3D можна легко перемикатися за допомогою наведення курсору миші на вікно перегляду і натискаючи клавішу Tab. Також, можна обрати 2D або 3D (або інші види, такі як зверху і спереду) з меню, що показує View Selection в правому верхньому куті вікна перегляду, як показано на рис 2.2.

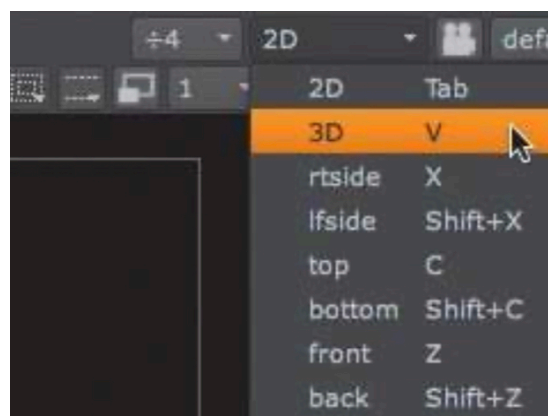


Рисунок 2.2 – Клавіша зміни видів у вікні перегляду

5. Наводимо курсор миші у вікні перегляду і натискаємо клавішу Tab для переходу в 3D режим (рис 2.3).

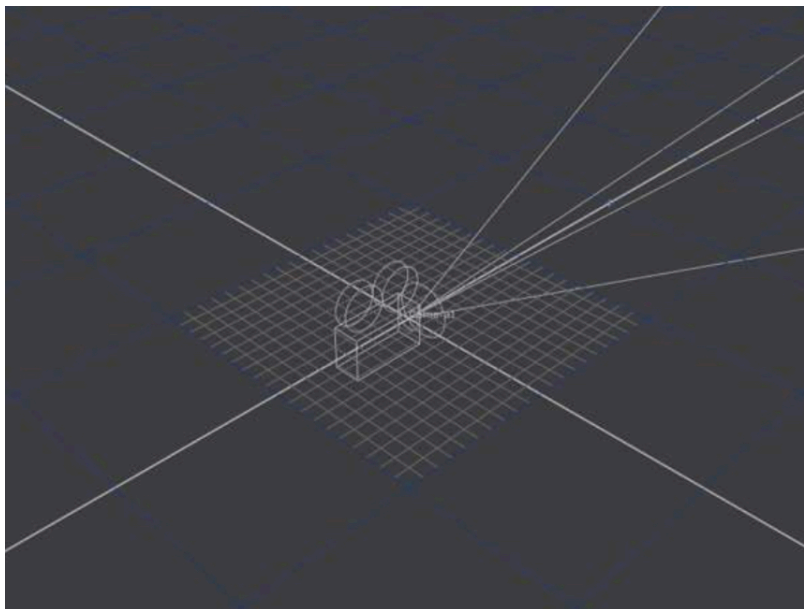


Рисунок 2.3 – Перегляд 3D камери в Nuke

Камера розташована посередині віртуального світу у “0, 0, 0” координатах. Всі ці 3D елементи формуються на цій площині. Саме цією камерою виконується перегляд 3D сцени.

## 2.2 Переміщення камери в 3D сцені

Наступні гарячі клавіші і методи використовуємо для навігації у 3D вигляді: Ctrl / Cmd + ліва кнопка на миші для повороту в сцені. Alt / Option + ліва кнопка на миші для переміщення у сцені. Колесо для прокрутки або клавіші +\/- для того щоб можна було зробити збільшення і зменшення масштабу. Переміщення камери в сцені робиться за рахунок переміщення її осі координат. Двічі клацнувши по камері активується вісь у режимі перегляду. Осі мають червоний, зелений і синій кольори, відображаються в центрі камери і дозволяють переміщати його (рис 2.4).

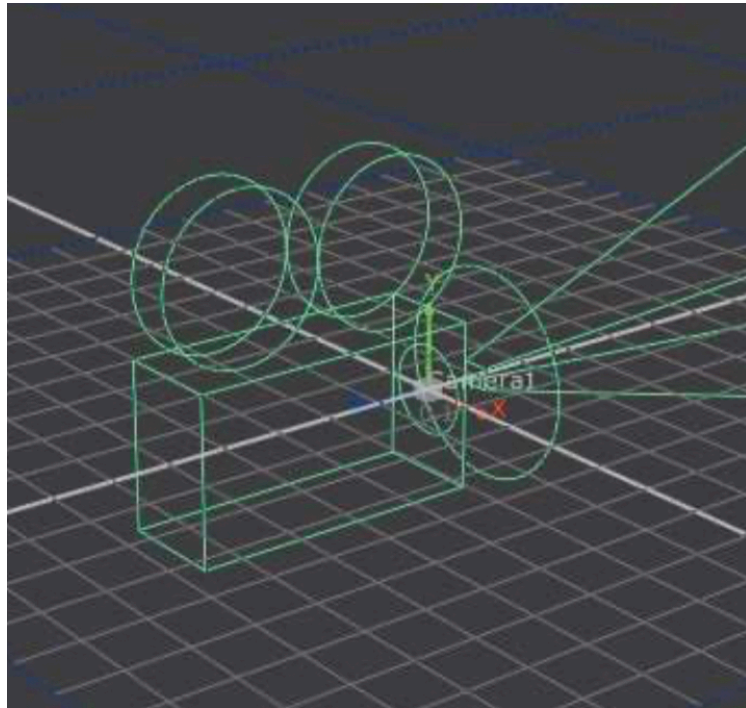


Рисунок 2.4 –Переміщення 3D камери в Nuke

Як правило, червоний, зелений, синій відповідає осям X, Y, Z. Іншими словами, червоний колір контролює напрямок X, зелений контролює напрямок Y, а синій відповідно Z. Така ж сама ситуація і для обертання камери. У вікні перегляду, утримуючи Ctrl / Cmd відобразиться елемент керування обертання (рис 2.5).

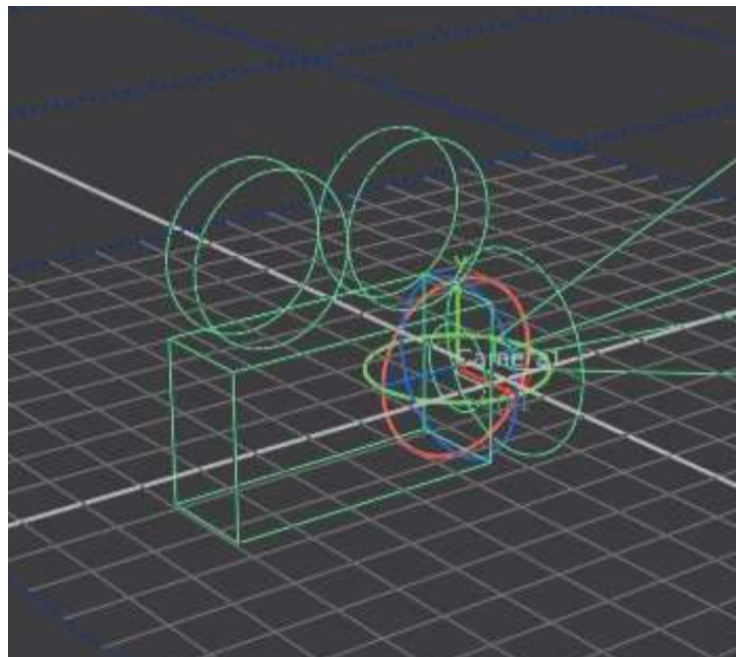


Рисунок 2.5 – Елемент керування обертання

## 2.3 Створення 3D об'єкта використовуючи пас світового положення

Пас світового положення - це пас, який рендериться в 3D програмному забезпеченні який не відображає візуальної частини зображення, але він має певні переваги. Пас світового положення відображає де знаходиться кожен піксель зображення відповідно до нульової координати (0,0,0). Використовуючи цей пас і ноду PositionToPoints можна створити хмару пікселів які будуть розташовані на своїх правильних позиціях в 3D світі[6].

1. У вікні перегляду, переключитися на перегляд безлічі каналів WorldPos (рис. 2.6).

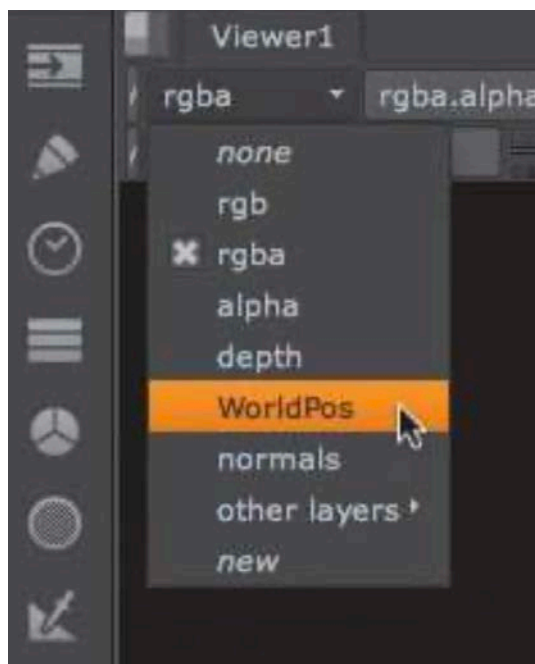


Рисунок 2.6 – Переключення на перегляд WorldPos

2. Навести показник на миші у вікно перегляду, в правому нижньому кутку побачити значення пікселів (рис. 2.7).

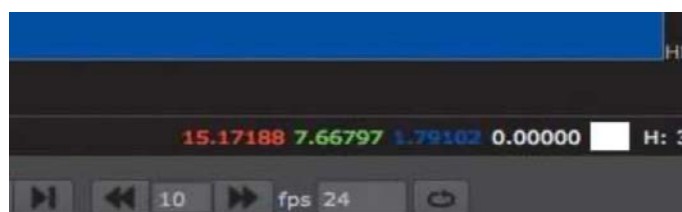


Рисунок 2.7 – Значення пікселів у вікні перегляду



Значення кольорів показують положення пікселів в 3D-просторі. Значення RGB відповідають координатам XYZ відповідно. Кольори не мають спільного з візуальною частиною зображення. Прохід світового положення створює дуже яскраве зображення.

Пас нормалей дуже схожий на пас світової позиції але з відмінністю у тому, що пас позицій являє собою пікселі в просторі, а нормалі являє собою кут пікселя в світових координатах.

3. Перейти на перегляд RGBA каналів у вікні перегляду[6].

4. З READ1 обраний в DAG, вставити PositionToPoints яка знаходиться в панелі інструментів 3D / геометрії. Глядач не перейде до 3D виду, так як більше не дивиться на зображення. Зображення, яке було у вікні перегляду тепер 3D-об'єкт, це буде видно якщо змінити пару властивостей[6].

5. На панелі PositionToPoints Properties треба обрати WorldPos з меню, що показує Surface Point. Для нормалей поверхні, обрати нормалі. Можна відразу побачити, що деякі пікселі з'явилися у вікні перегляду. Це насправді не пікселі, але малі 2D площину, як шматочки паперу, що плавають в 3D-просторі[6] (рис 2.8).

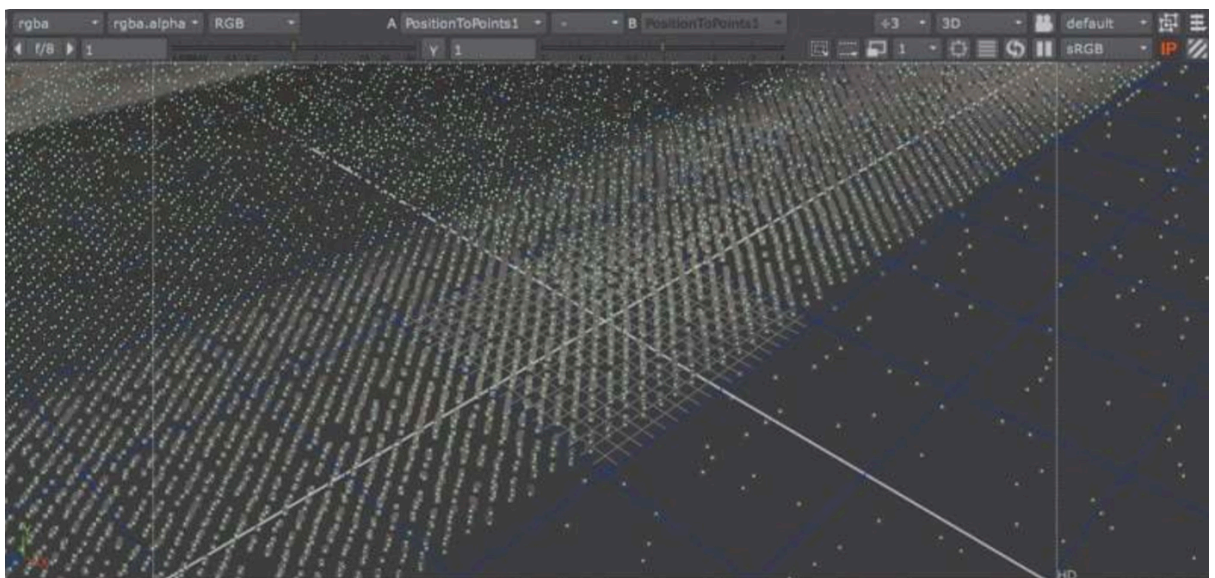


Рисунок 2.8 – Показ пікселів в 3D-просторі



Переміщуючи контролер в 3D просторі, можна побачити зображення у дуже гарній якості. Але в будь-якому випадку, це зображення повинно бути видно з конкретної камери.

## 2.4 Імпорт камери

Для імпорту камери зі сторонніх джерел необхідно виконати такий алгоритм дій. При виконанні 3D композингу необхідно мати камеру, яка б вела себе аналогічно до тієї, яка знімала цю сцену. Перш за все, необхідно налаштувати 3D сцену.

1. Не вибирати нічого в DAG, обрати 3D / Налаштування на панелі ІНСТРУМЕНТІВ (рис. 2.9).

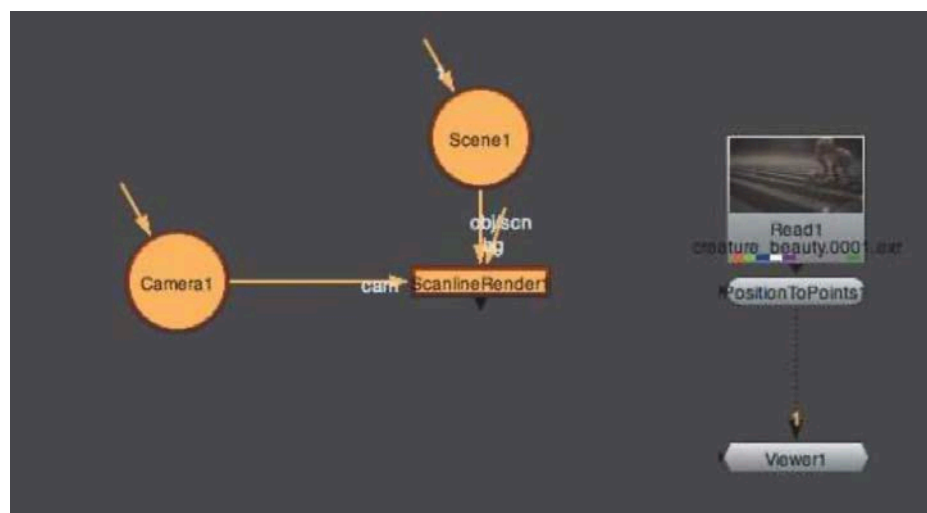


Рисунок 2.9 – Набор інструментів та його використання.

2. Клацнувши двічі на CAMERA1, ждя того щоб відобразити її панелі властивостей. Найвища властивість називається читанням з файлу. Вона необхідна при використанні вкладки Файл, коли виконується читання розташування камери і інші властивості з файлу (рис. 2.10).

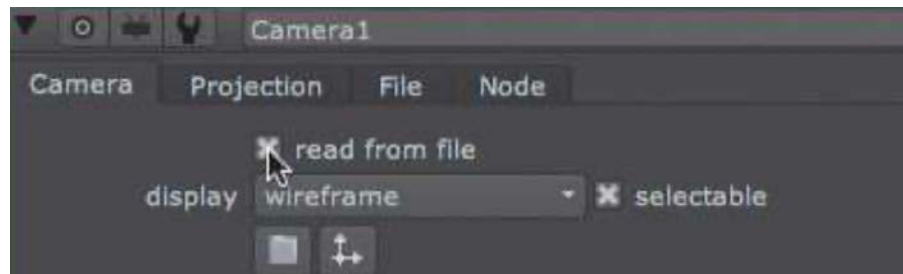


Рисунок 2.10 – Дозволяє читати Файл при натисканні

3. Обираємо Читання з файлу.
4. Переходимо на вкладку Файл у панелі властивостей CAMERA1 (рисунок 2.11).

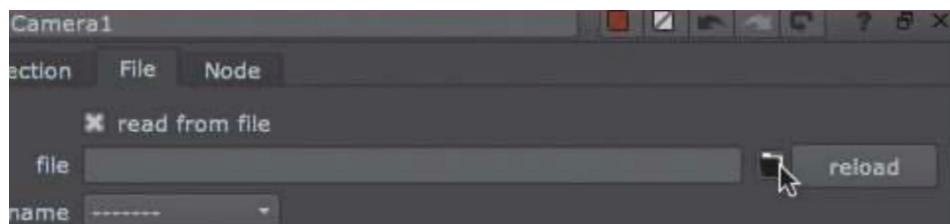


Рисунок 2.11 – CAMERA1 у вкладці Файл

5. Клацаємо на значок папки праворуч від властивості файлу, щоб завантажити браузер файлів, для імпортування файлу камери.
6. Переходимо в каталог з Camera.fbx у браузері файлів.
7. З'являється діалогове вікно із запитом, щоб знищити дані поточної камери; натискаємо кнопку Так.

Файл FBX може містити багато властивостей камер в різних дублях наприклад з двох розкритих меню, необхідно обирати ім'я вузла

8. Обираємо Take 001 з меню, що випадає Take Name.
9. З того меню, що випадає Ім'я вузла, обрати S26A\_Camera (рис. 2.12).

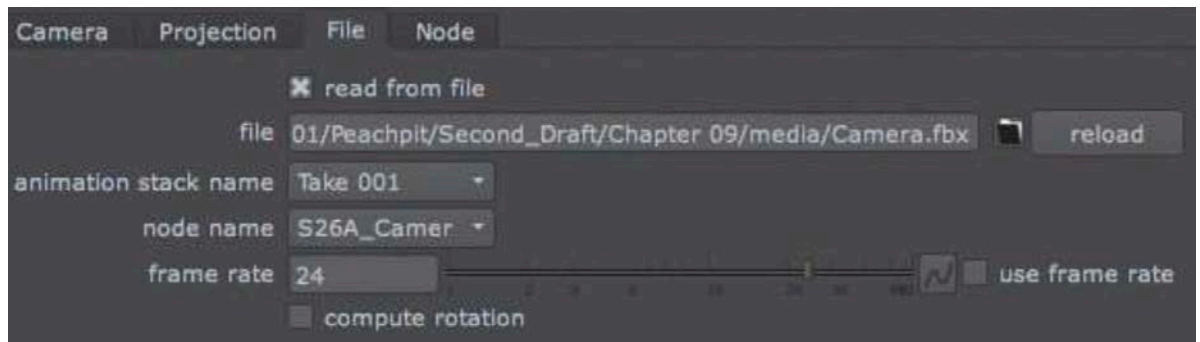


Рисунок 2.12 – Меню імпорту камери

10. Перейти у вкладку Camera.

Можна побачити, що поля переміщення і повертання вже заповнені значеннями, і вони неактивні та недоступно для зміни. Таким чином здійснюється анімація камери. Якщо у файлі на диску зміняться значення, то в цих полях також зміняться. Камера зникає з вікна перегляду. Для того, щоб її побачити знову, треба зробити такі дії:

11. Обираємо CAMERA1 у вузлі Graph.

12. Навести курсор миші на вікно перегляду і натиснути клавішу F, щоб обрізати перегляд для обраного елемента (рис. 2.13).

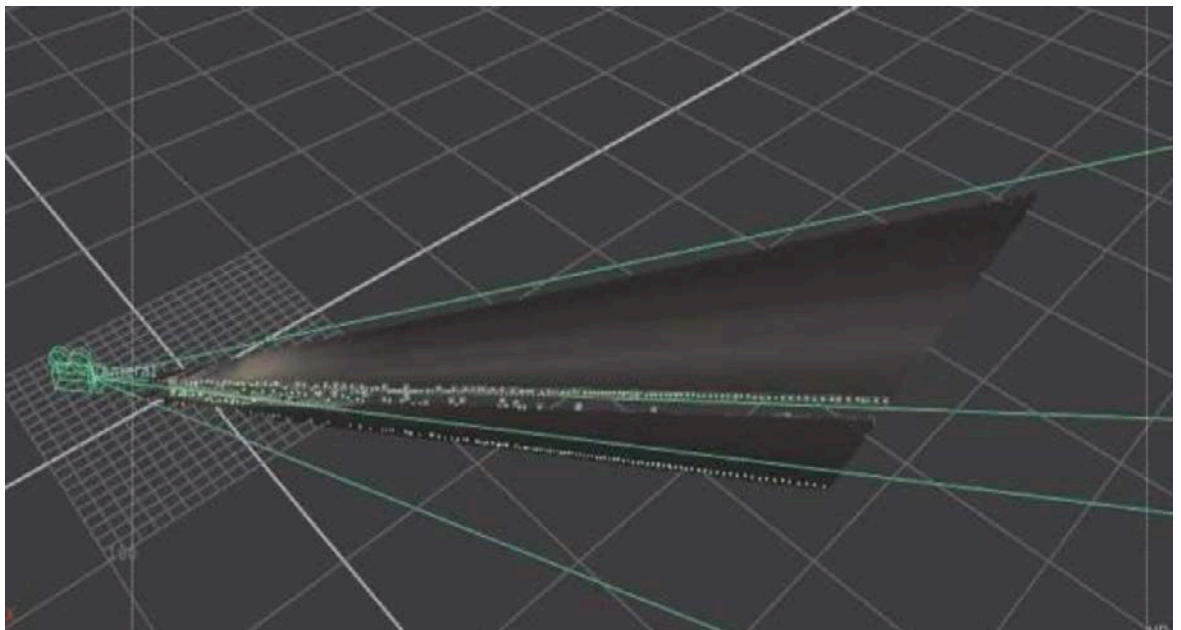


Рисунок 2.13 – Вид камери після проведення імпорту

Також є можливість візуально побачити зв'язок між камерою і точками, створеними з вузлом PositionToPoints. Ці частинки існують тільки в області поля зору камери. Для перегляду сцени з імпортованої камери.

13. Обрати на рис.2.14 CAMERA1 з перегляду камери в спадному меню в правому верхньому куті вікна перегляду.

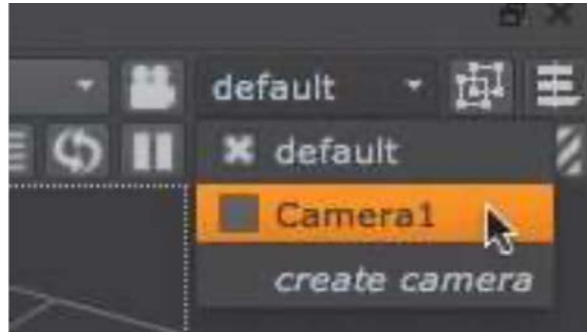


Рисунок 2.14 – Вибіраємо камери для перегляду

Вид точки зображення під правильним кутом з CAMERA1. (рис. 2.15).



Рисунок 2.15 – Зображення з складених точок

На зображенні можна побачити недостаючи компоненти, але це нормально бо за допомогою цього методу можуть бути відтворені точки які знаходяться в полі зору камери. Потім отриману інформацію з точок можна використати для розташування об'єктів у просторі.

## 2.5 Імпорт системи частинок 3D ДЕРЕВА NODE

Система частинок нод зберігається як інший сценарій Nuke, та його можна об'єднати за допомогою команди Import Script в меню Файл.

1. З меню Файл обрати команду Імпорт сценарію.
2. Перейти в папку з частинками і потім двічі клацнути по Particles.nk.
3. В програмі Nuke, можна отримати повідомлення про те, що ви використовуєте інструменти NukeX. (рис. 2.16).



Рисунок 2.16 – Запуск інструментів NukeX в Nuke генерує це повідомлення

NukeX не може мати такого повідомлення і тому можна змінити налаштування будьякого з вузлів частинок пізніше. Імпортовані вузли можна з легко переміщати тому, що вони одразу являються обраними в сцені.

4. Переміщення вузлів з розташуванням інших вузлів (рис. 2.17).

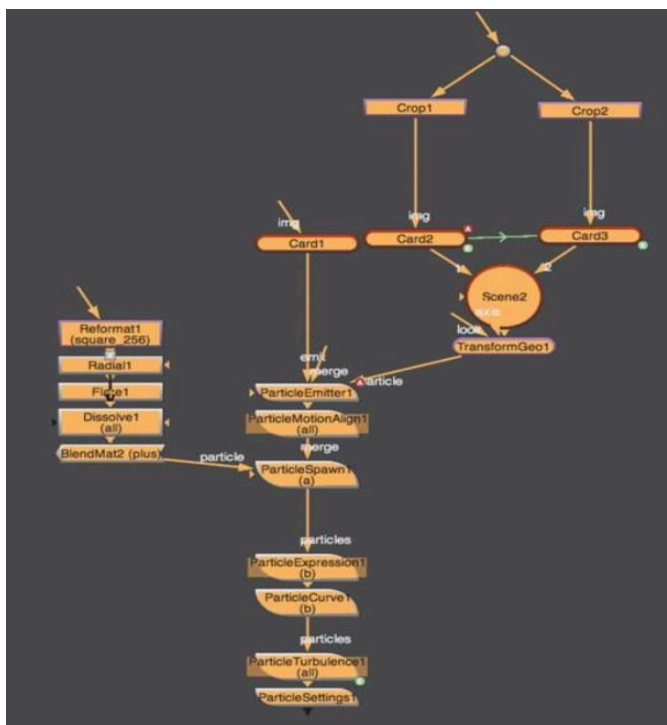


Рисунок 2.17 – Дерево вузлів що генерує частинки

В даному дереві вузлів, не вистачає одного елемента, це зображення частинки.

5. Зверху над деревом яке ми імпортували, є вузол точки над двома вузлами. Підключити вихід read2 до входу цієї точки (рис 2.18).

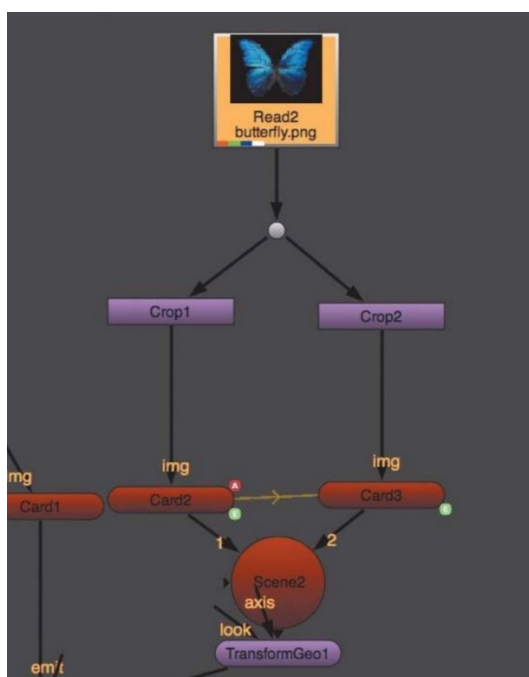


Рисунок 2.18 – Підключення зображення метелика

Основні властивості і можливості дерева вузлів для генерації частинок в 3D просторі програми Nuke.

- Два данні вузли Crop під Read3 розділяють метелика на ліву половину і праву половину.

- Card2 і Card3 дві плоскі 3D поверхні. До цих нодів підключені вирази, які змушують їх рухатися в синхронному порядку. Вони являють собою паперовий тонкий об'єкт в 3D-просторі. Зображення метелика (ліве і праве) текстури цих поверхонь[6].

- СЦЕНА2 з'єднує дві частини метелики і TransformGeo1, який являє собою 3D-вузол трансформації, який дозволяє обертати комбінацію з двох вузлів карти, щоб краще відображалися перед камерою[6].

6. Для перевірки треба в вікні перегляду необхідно двічі клацнути TransformGeo1, та натиснути 1[6].

- Card1 є джерелом частинок, що випромінюються. Ще його називають випромінювачем або пістолетом.

- ParticleSettings1 останній вузол в цьому дереві. Він являється виходом з цього дерева.

7. Щоб зробити перегляд анімації руху метеликів необхідно натиснути Play (рис. 2.19).

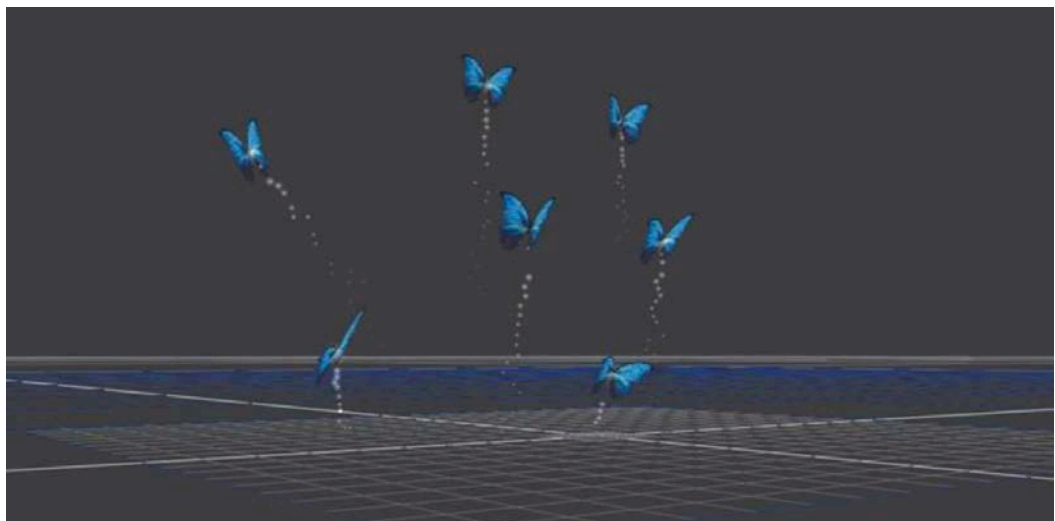


Рисунок 2.19 – Анімація метеликів створена за допомогою системи частинок

## Висновки до розділу 2

В другому розділі дослідження наведені наочно основні проблеми, які пов'язані зі створенням 3D-композиції в програмному забезпеченні Nuke. Також окремо, наведено ще такі операції, як вибір камери в 3D режимі та інсталяція сцени. Побудував модель в середовищі Nuke, яка дозволяє розробникам в сфері кіно провести тестування можливостей програми при створенні віртуальних ефектів у сучасних фільмів.

### **3 СУМІЩЕННЯ ЕФЕКТІВ У ПРОГРАММІ NUKE**



3D простір та його можливості програмного середовища Nuke були зроблені для спрощення зв'язку композитингу з 3D компонентами. Ще 3D модуль Nuke використовують, щоб легше виконувати 2D трекінг інформації, спростити ротоскопінг і багато, багато іншого. Крім цього, якщо в сцені наявні 3D об'єкти, композер може отримати дані 3D трекінгу для використання у своїх цілях. Nuke добре працює із зовнішніми 3D-додатками і може обмінюватися даними з ними, використовуючи формати файлів, зазвичай використовуваних в 3D-додатках: Alembic, OBJ і FBX. Розширення .obj для файлів, які мають лише геометрію. Alembic і FBX типи файлів можуть містити практично все, що генерується в 3D сцені.

### 3.1 Суміщення 3D-об'єктів в просторі

Вузол TransformGeo так само, як і вузол 2D Transform дозволяє переміщувати об'єкти в просторі. Єдина відмінність полягає у тому, що він призначений для переміщення об'єктів або геометрії, але не для 2D-зображень[6]. Це також означає, що він рухається не тільки по двох осях, як вузол 2D Transform, але по трьох. Для суміщення 3D- об'єктів необхідно:

1. Обрати ParticleSettings1 і вставити вузол TransformGeo після нього. Даний вузол створює вісь, яка переміщує всю систему частинок в 3Dпросторі. Можливо використовувати осі для переміщення, і утримуючи Ctrl / Cmd, для повороту. Елементи управління є такими ж, як для камери[6].

2. Двічі клацнути PositionToPoints1 і CAMERA1[6].

3. Потрібно обрати CAMERA1 з меню перегляду камери в правому верхньому куті.

4. Наступним обираємо TranformGeo2 в DAG. Відтепер можна бачити вісь управління системою частинок в засобі перегляду відносно істоти (рис 3.1).

Наступним кроком необхідно помістити систему частинок саме там, де голова істоти знаходиться між прутами решітки.

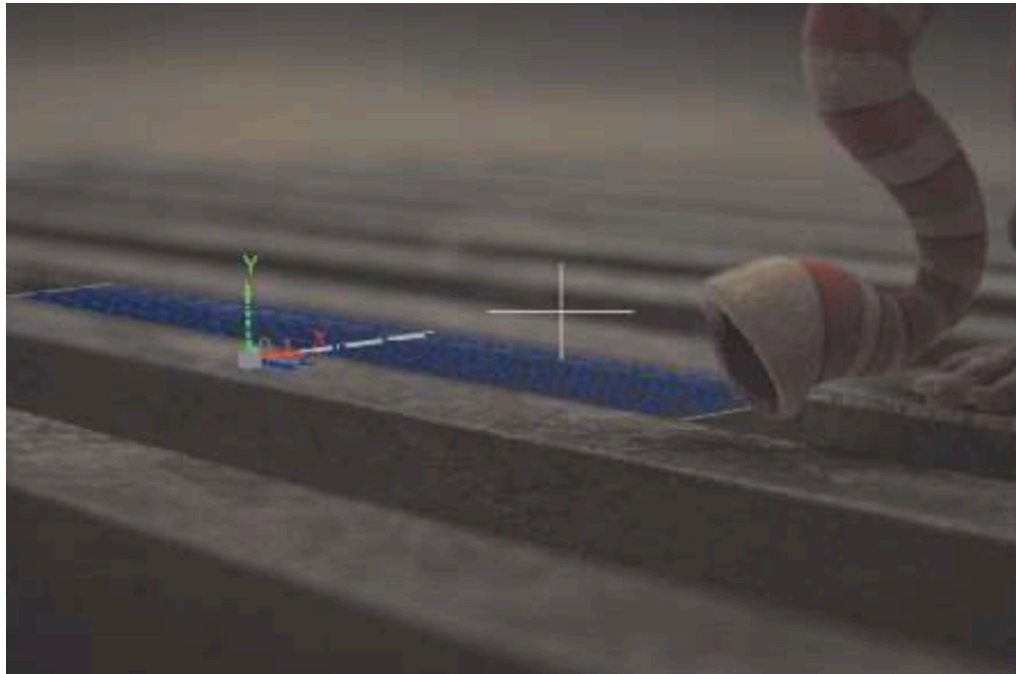


Рисунок 3.1 – Положення вісі частинок відно 3D простору.

5. Перейти до кадру 40 в TimeBar. Голова цієї істоти вставляється в цю решітку. Необхідно розташувати TransformGeo2 прямо в цьому кадрі.

6. Перемістити TransformGeo2 у місце, де знаходиться голова істот. Такі координати голови істоти  $X = 2.2$  і  $Z = 11,7$ . Проте, метелики повинні випускатися з-під ґратки, тому необхідно перемістити систему частинок вниз.

7. Використання панелі властивостей TransformGeo2 для зміни Y координати на -2.

8. Перейти до кадру 90 в TimeBar. За замовчування елементи системи частинок досить малі, тому щоб їх було добре видно в кадрі необхідно збільшити їх розміри.

9. Повернутися до кадру 40 в TimeBar.

10. У панелі властивостей TransformGeo2, змінюємо Uniform Scale 5. Результат зміни розміру частинок можна побачити на (рис 3.2).

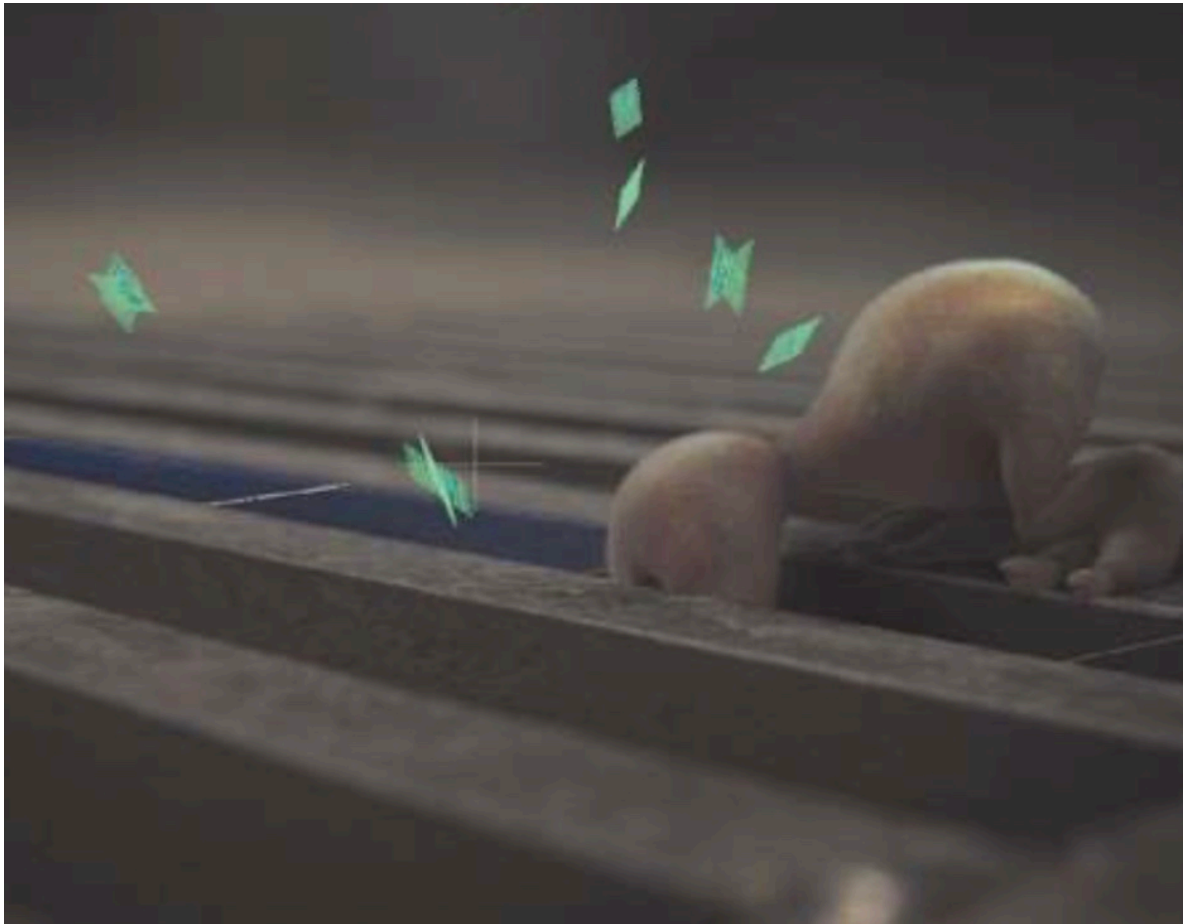


Рисунок 3.2 – Метелики в правильному місці і розмірі.

### 3.2 Переміщення 3D-об'єктів в часі

Тепер необхідно розташувати систему частинок в правильному місці в часі. Всі вузли пов'язані з часом знаходяться в Time інструментів. Деякі з цих вузлів не тільки працюють над кліпами зображень, вони також працюють з геометрією. Вузли, які діють на геометрію FrameRange, Offset і FrameHold. Для переміщення 3D фігур в часі застосовують Offset[6]. Обрати TransformGeo2 в DAG і вставити вузол Offset після нього від часу набору інструментів (рис 3.3).

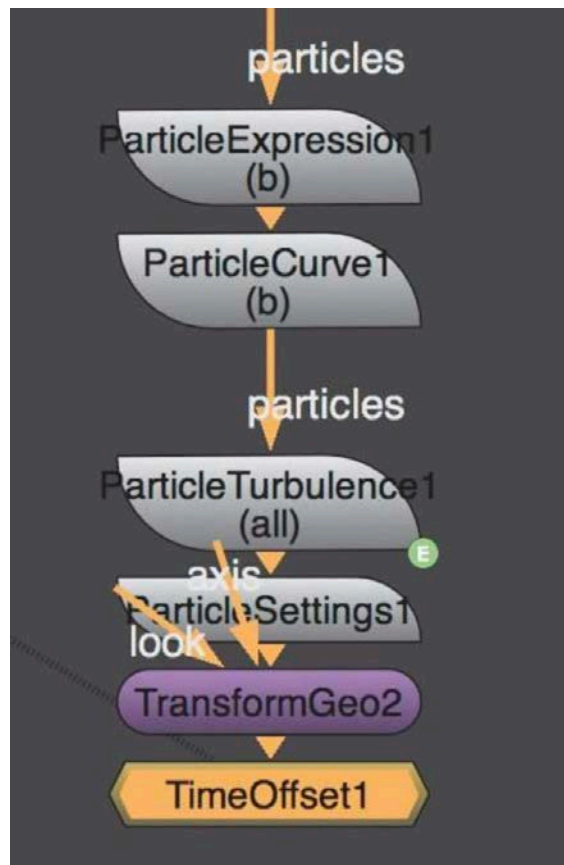


Рисунок 3.3 – Offset вставляється після TransformGeo2

Слідуючим кроком необхідно перемістити початок створення частинок на 30 кадрів вперед. Для цього в панелі властивостей TimeOffset1, в розділі Time Offset необхідно задати значення введіть 30 в поле введення. Тепер елементи частинок будуть створюються в правильному часі[6].

### 3.3 Переведення 3D об'єктів в 2D пікселі

В даний момент в полі DAG існує три основні дерева композиції. Для зручності вони згруповані у фігури як показано на рис. 3.4.

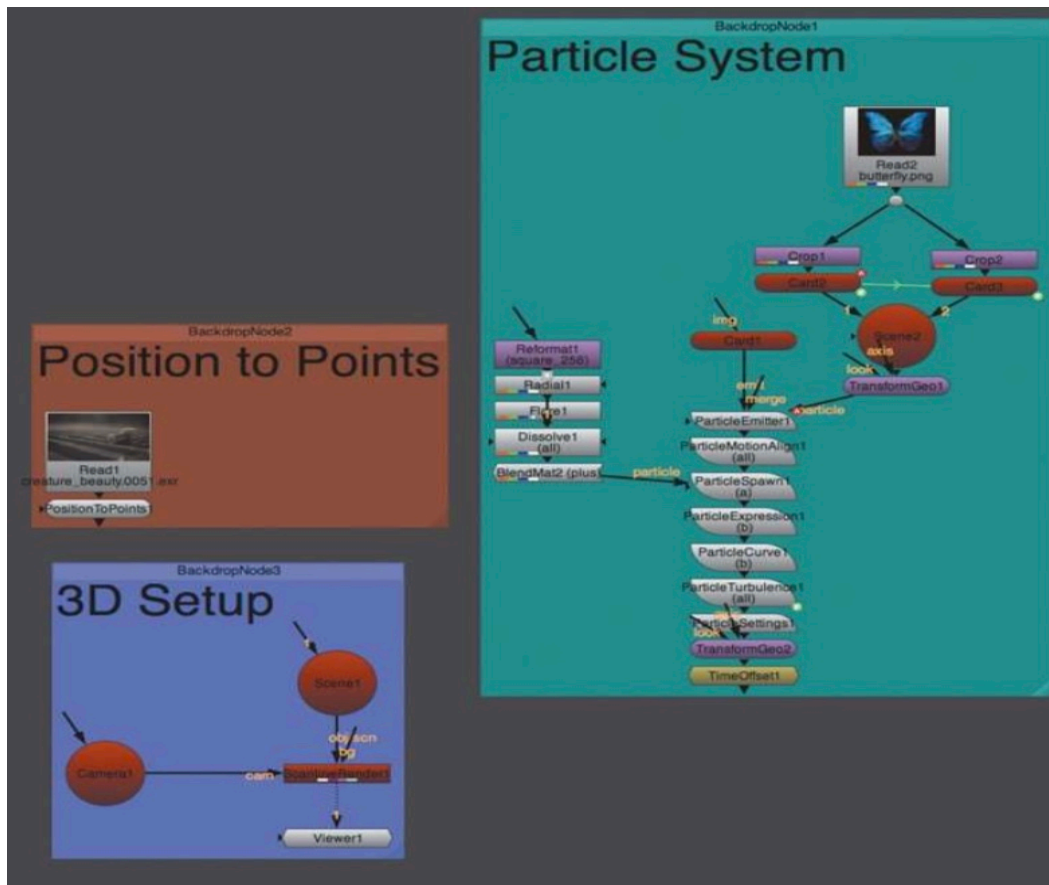


Рисунок 3.4 – Три дерева вузлів

Область по праву сторону це система частинок. Область зліва є посилення на об'єкт що генерується вузлом PositionToPoints. Область яка знаходиться в нижній частині є налаштуванням 3D сцени. Наявність окремих дерев дає багато свободи дій і гнучкості. Для підключення частинок необхідно[6]:

1. Перетягнути вихід TimeOffset1 на вході СЦЕНИ1.
2. Обрати ScanlineRender1 і натиснути 1, щоб переглянути його у вікні перегляду.

Вузол ScanlineRender виконує рендеринг 3D сцени за таким же принципом що і зовнішнє 3D програмне забезпечення, 3D-геометрію за допомогою камери і 2D-зображення, яке можна маніпулювати в Nuke з будь-яким 2D вузлом, як Merge, Grade і Blur.

Можна виконати композитинг композиції з метеликами поверх зображення фону, це було б не можливо якщо б не перетворили геометрію в 2D зображення.

3. Обравши ScanlineRender1, натиснути М для вставки вузла Merge.

4. Підключити вхід Merge1 в В для READ1 і перегляд Merge1 у вікні перегляду.

5. Повернутися до кадру 1 і натиснути кнопку відтворення, щоб подивитися систему частинок, як 2D-зображення (рис 3.5).

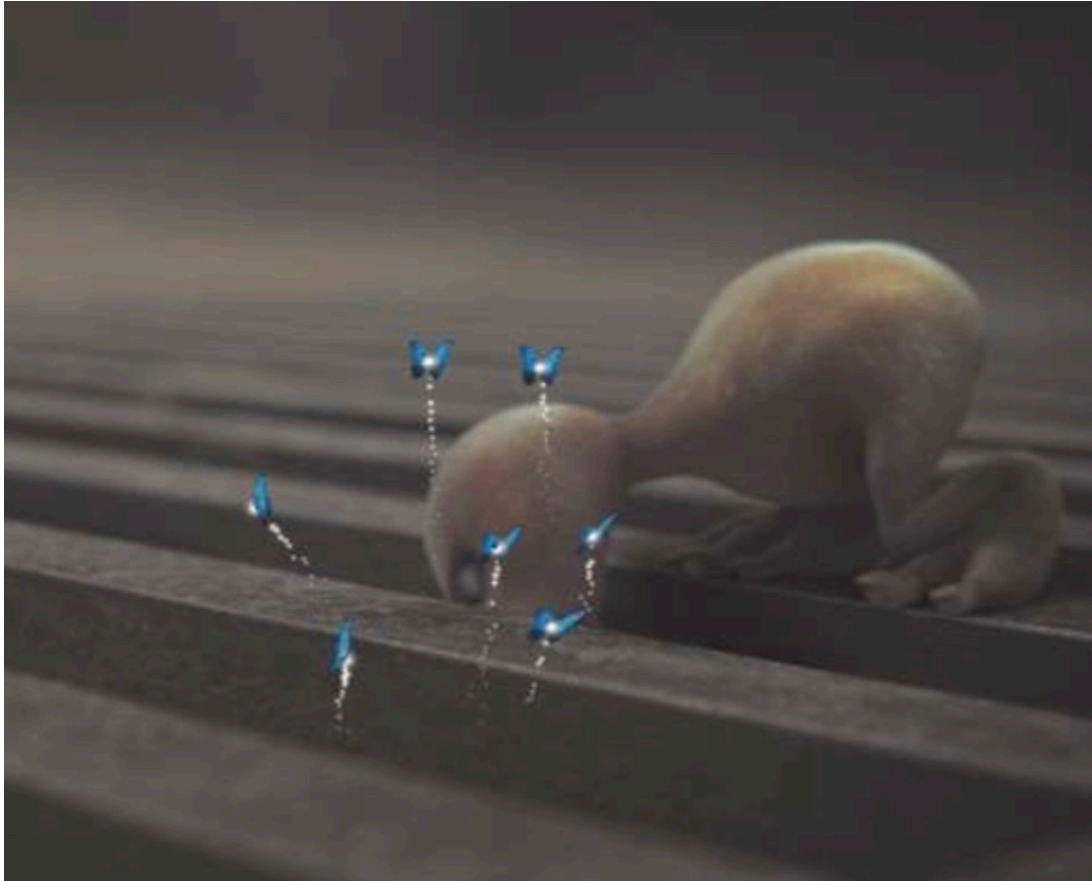


Рисунок 3.5 – Метелики як 2D-зображення, скомпоновані над фоном

В цьому зображенні необхідно виправити три пункти. По перше метелики мають зовсім іншу кольорову гаму зображення. По друге вони мають з'являтися з під ґратки люку. І по третє деякі з них мають летіти за істотою. Для виправлення цих недоліків необхідно:

6. Натиснути кнопку Стоп і перейти на кадр 50.

7. Обрати нічого, натиснути Р, щоб створити вузол RotoPaint.

8. Помістити курсор миші у вікні перегляду і натиснути V, щоб почати малювати фігуру Безьє. Намалювати уздовж решітки, як показано на рис 3.6.

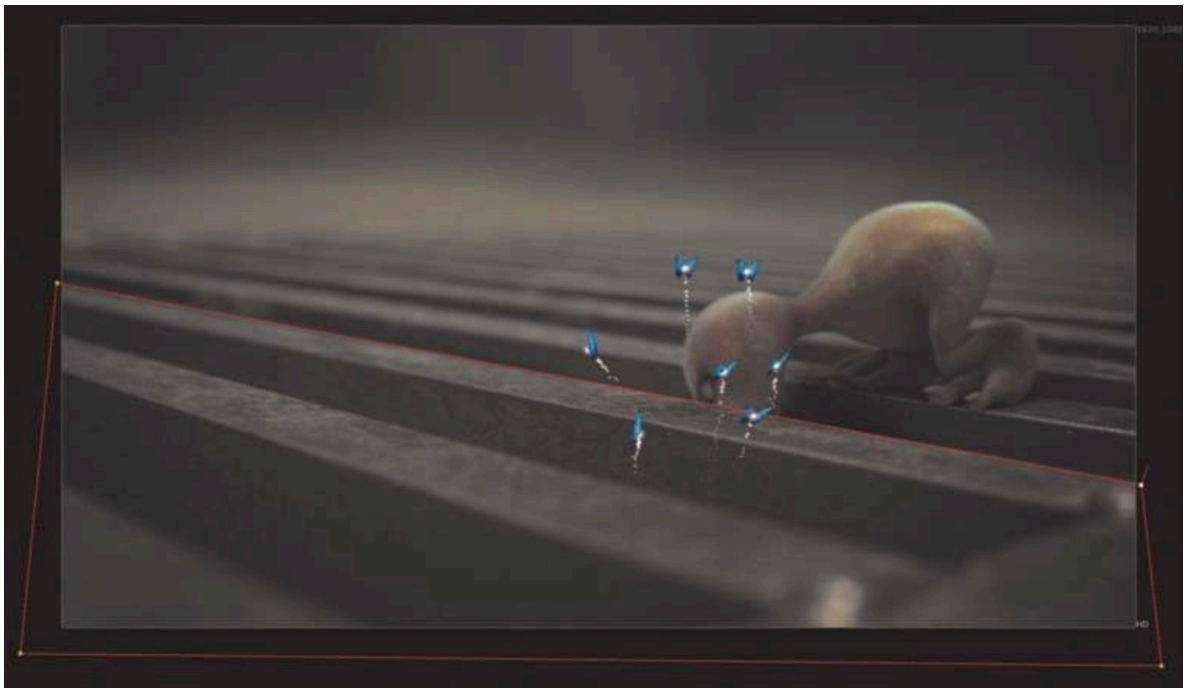


Рисунок 3.6 – Створена маски вузлом RotoPaint.

Тепер застосовуємо RotoPaint вузол з маскою, для маскування метеликів.

9. Обрати RotoPaint1 натиснути М, щоб вставити інший вузол Merge.

10. Помістити Merge2 зв'язку між ScanlineRender1 і Merge1.

11. У меню, операцій що випадає Merge2, обрати команду трафарет.

Рис. 3.7 показує метеликів які маскуються за ґраткою і вже здається, що вони надходять з-під підлоги. Край маски є трохи різким.

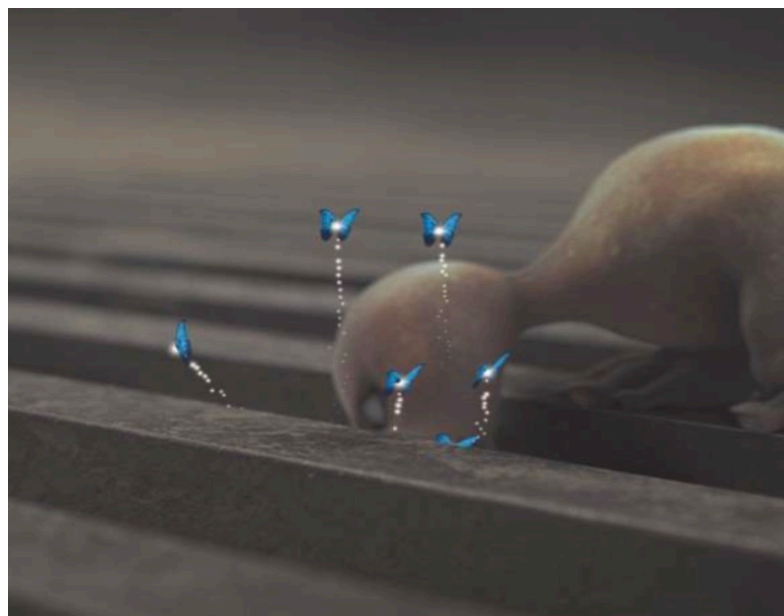


Рисунок 3.7 – Замасковані метелики



12. Обрати RotoPaint1 і натиснути на кнопку В, щоб вставити вузол Blur.

13. Встановити значення Blur1 на 3 пікселі. Тепер наш край зображення не виглядає таким різким. Тепер дерево нодів даної композиції виглядає так як показано на рис. 3.8.

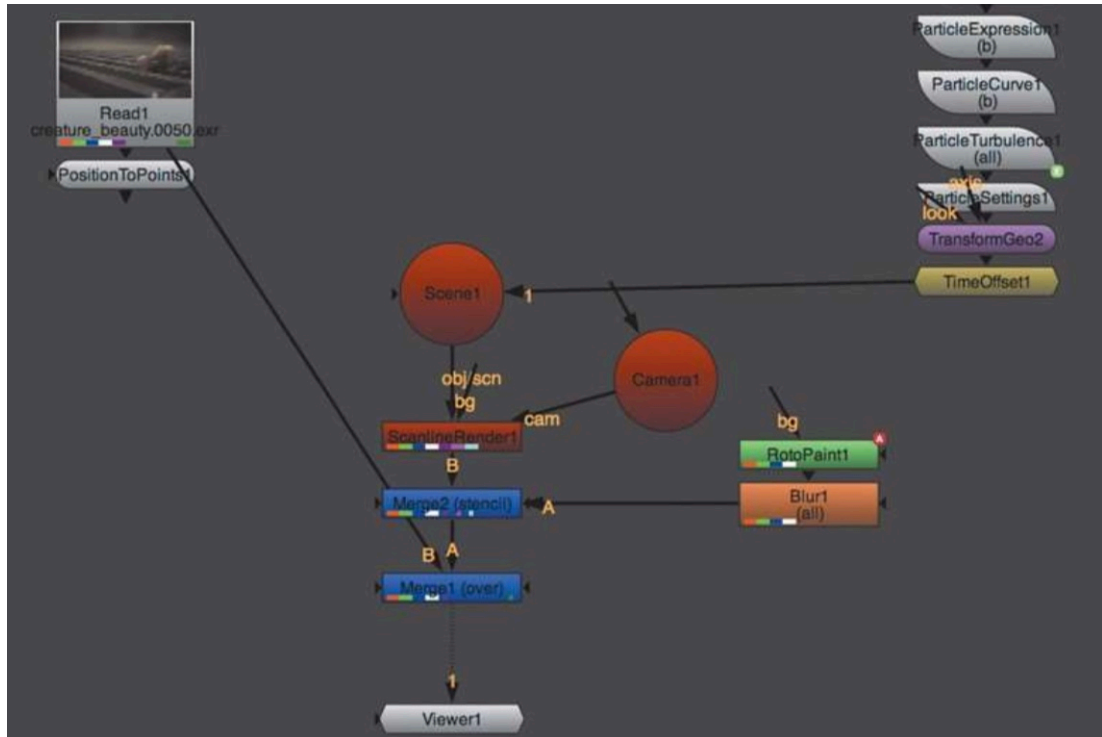


Рисунок 3.8 –Дерево нодів після маскування метеликів

### 3.4 Застосування матеріалів до об'єктів

Помістити частину метеликів перед об'єктом, а іншу частину за об'єктом є досить складною задачею. Для цього необхідно виконати наступне. В сцені вже існує об'єкт де сидить істота і його координати можуть бути використані в якості маски-істоти зроблені з точок. Для цього:

1. Підключити вихід з PositionToPoints1 до СЦЕНИ1 (рис 3.9).



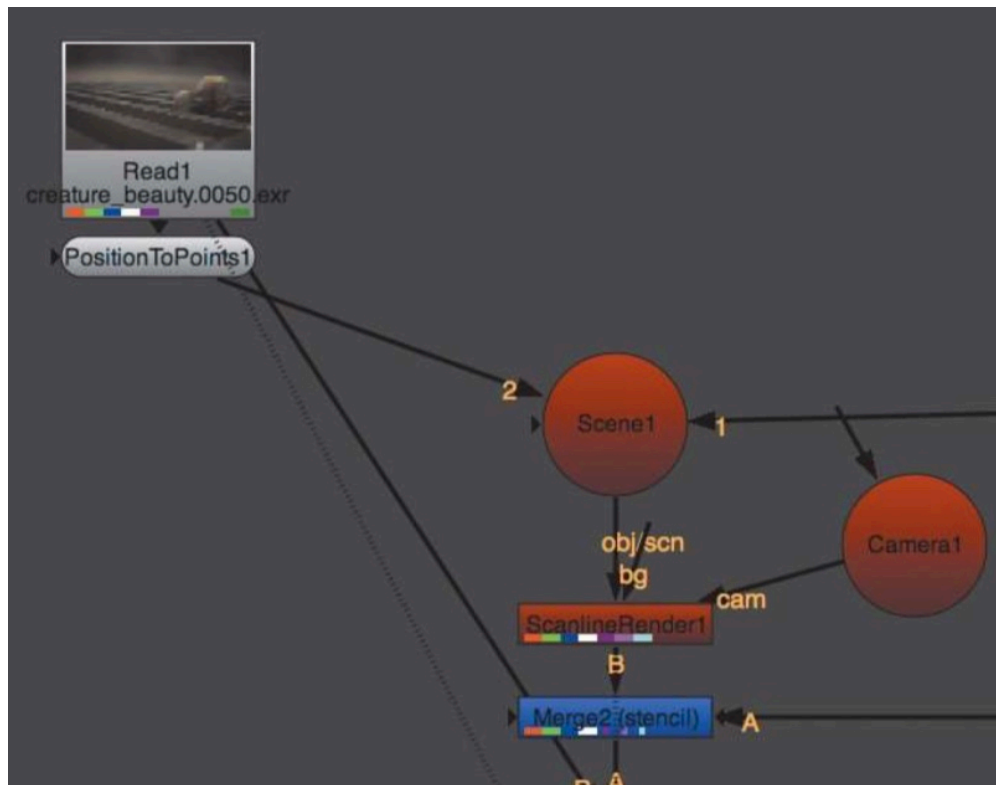


Рисунок 3.9 – Підключення іншого об'єкта до сцени

Тепер, коли PositionToPoints1 підключений до СЦЕНИ1, як точки об'єкта істоти і система частинок існує в тому ж просторі, вони будуть рендеритися разом через вузол ScanlineRender.

2. Обрати READ1 і натиснути 2, щоб переглянути його в другому вході Viewer.

3. Обрати ScanlineRender1 і натиснути 1, щоб переглянути його в першому вході Viewer.

4. Перемикання між входами 1 і 2 відбувається при наведенні покажчика миші у вікно перегляду і натисканні клавіш 1 і 2.

Після виконання цих дій деякі з метеликів дійсно зникають позаду голови істоти. Але виникає одна проблема, що все відображене на вході 1 є товстішим ніж потрібно. Це відбувається тому, що вузол PositionToPoints не дає точної геометрії. Точки, створюють не правильний розмір об'єктів. Можна отримати точніше зображення, використовуючи Point Size властивість PositionToPoint, але PositionToPoints не розраховані і не призначені для цього. Можна і надалі

використовувати необроблений READ1 в якості фону, але потрібно використовувати PositionToPoints1 як маскою в 3D-просторі[6].

Об'єкт, як правило, має матеріал, який ми приміняємо до нього. Вузли карт, складові метелики мають зображення метелика покладені на них в якості матеріалу; зображення метелика підключене до входу Img вузлів карт (рис. 3.10). Якщо не призначити матеріал, то на них буде пусте зображення, що можна побачити в 3D-перегляді, але нічого не буде рендеритися через вузол ScanlineRender.

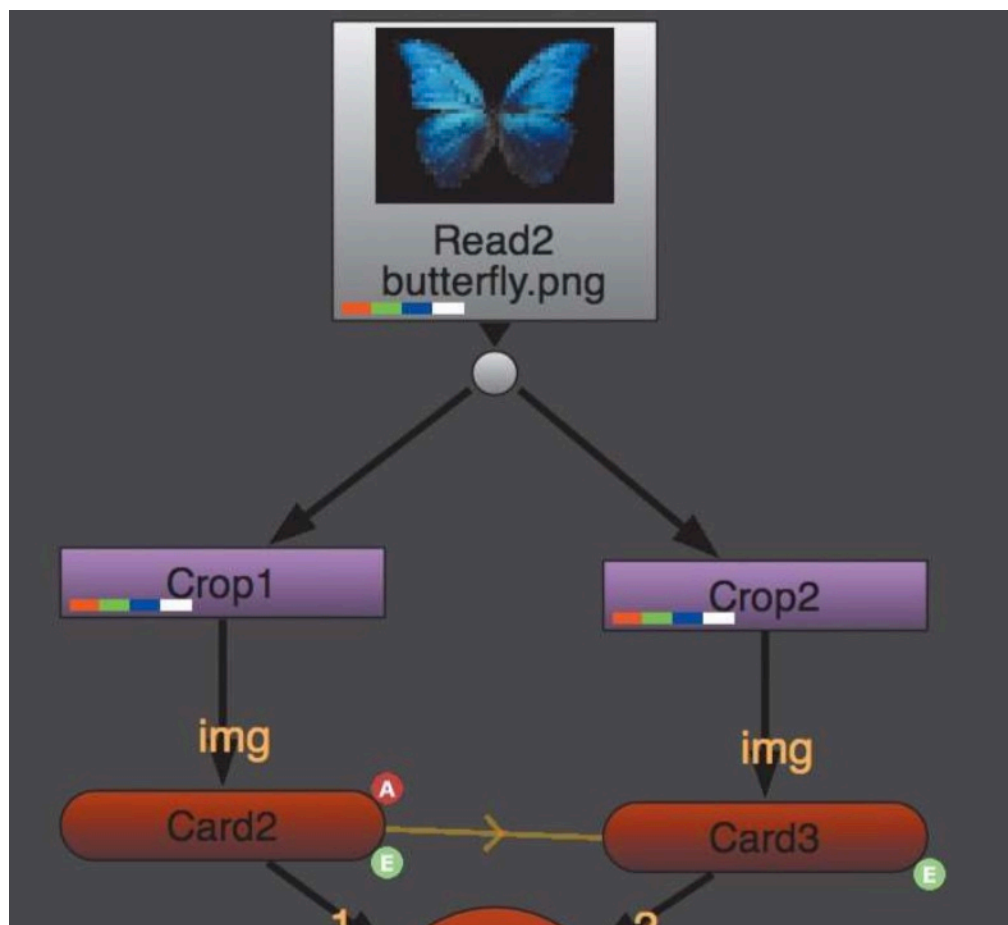


Рисунок 3.10 – Ноди зображень підключені в якості матеріалів

Крім простих матеріалів, таких як зображення, деякі більш складні матеріали мають більший контроль над тим, як вони реагують на світло; вони можуть бути блискучими або випромінювати світло, і вони можуть контролювати

колір своєї тінь, наприклад. Всі ці матеріали можна знайти в 3D> Shader (Шейдер це ще один термін для матеріалу).

Одним з таких матеріалів називається FillMat. Сенса цього матеріалу або шейдера повернути геометрію до якої він призначений у маску для іншої геометрії в сцені. Оскільки PositionToPoints не має входу Img, необхідно використати інший вузол з 3D> Shader інструментів під назвою ApplyMaterial.

5. Обрати PositionToPoints1 і з панелі інструментів 3D> Shader обрати ApplyMaterial.

6. Обирати нічого і створити FillMat з 3D> Shader.

7. Підключити вихід FillMat1 до Mat входу ApplyMaterial (рис. 3.11).

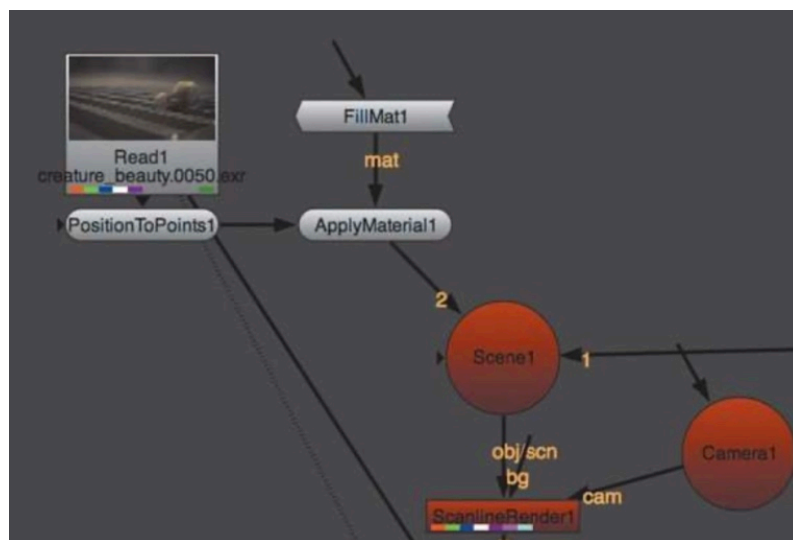


Рисунок 3.11 – Підключення матеріалу за допомогою вузла ApplyMaterial

8. Перегляд ScanlineRender1 у вікні перегляду (рис 3.12).



Рисунок 3.12 – Істота маскує деяких метеликів

В даний момент не видно істоти на виході ScanlineRender1.

9. Перегляд Merge1 у вікні перегляду (рис. 3.13).

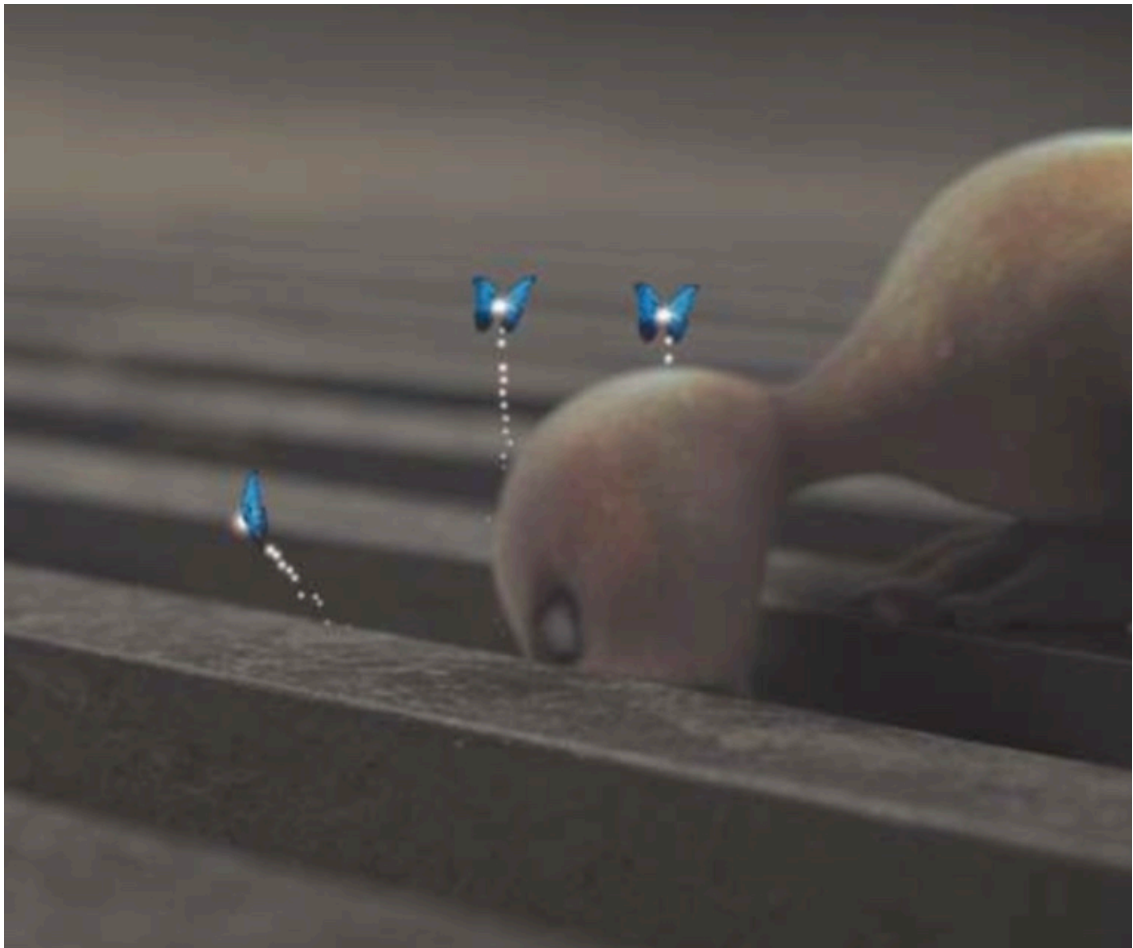


Рисунок 3.13 – Вигляд композиції з маскою істоти

### 3.5 Корекція кольору зображення

Таким чином, вирішені майже всі проблеми 3D-композиції. Далі необхідно зробити метеликів максимально наближеними по відношенню кольору сцени і зробити їх не такими різкими, для цього:

1. Вставити вузол Grade між ScanlineRender1 і Merge2.
2. Для входу ScanlineRender1 обрати RGBA.alpha з (Un) premult.
3. Змінити колір метелика на більш теплий і можливо, трохи яскравіший.

Вихідні значення:  $R = 1,523$   $G = 1,24$ ,  $B = 0,77$ .

4. Вставити вузол Blur після Grade1.
5. В панелі властивостей Blur2, змінити розмір до 4.

6. Закрити всі панелі властивостей, натиснути кнопку відтворення у вікні перегляду. (рис. 3.14).

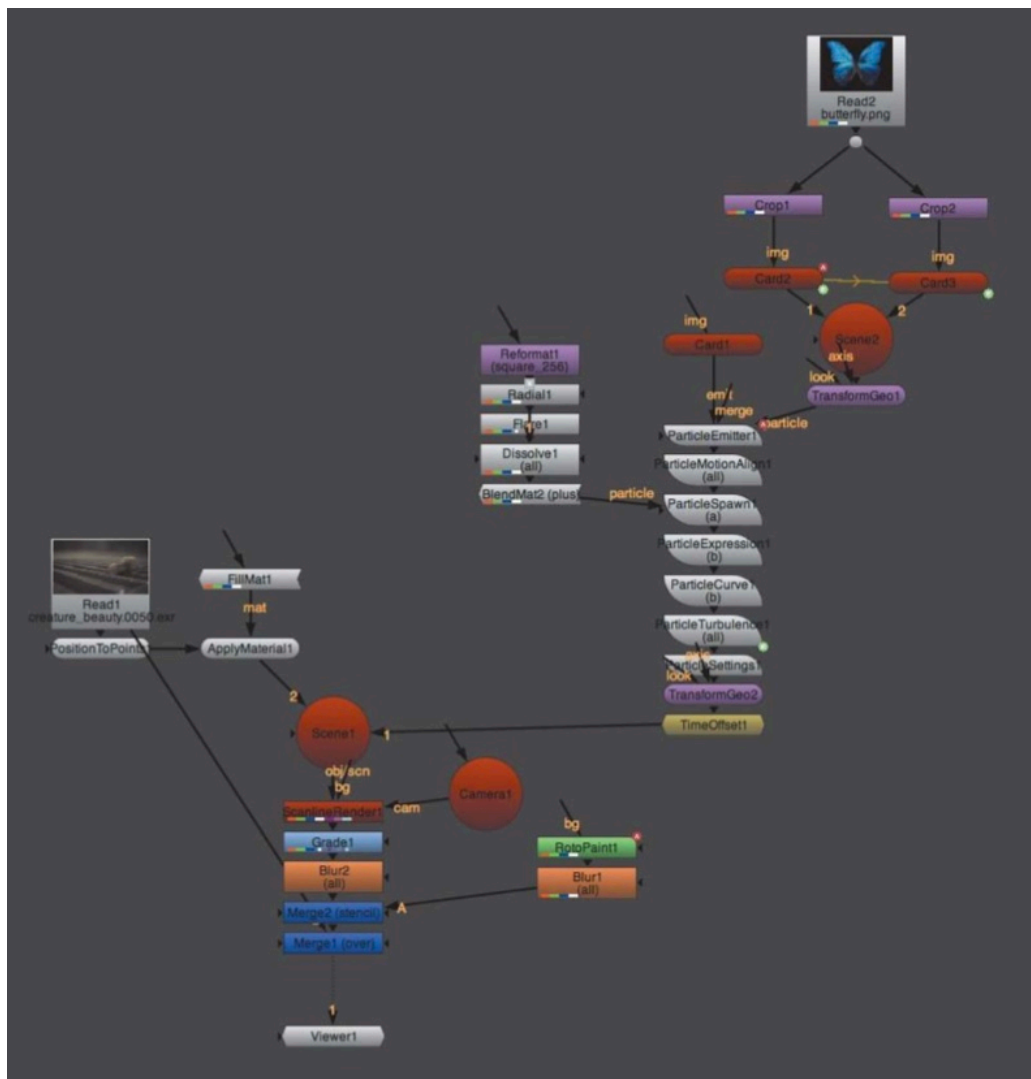


Рисунок 3.14 – Готове дерево нодів 3D сцени

### **Висновки до розділу 3**

Проаналізовано основні функції та можливості нодів, які відповідають за рух в сцені, суміщення 3D об'єктів в просторі. Показано у детальному розрізі весь алгоритм дій таких процедур як, наприклад, переведення 3D-об'єктів в 2D пікселі. Зокрема, визначено такі характерні особливості нодів, які визначають створення ефекту переміщення метеликів та накладання маски.

## 4 РОБОТА З ВІРТУАЛЬНИМ ЕФЕКТОМ ВИБУХУ В ПРОГРАМІ HOUDINI

Досить часто, при створенні ефектів для кіно, реклами, різного роду роликів, використовують програму Houdini. Компоненти цієї програми дозволяють ефективно пов'язувати різні елементи і об'єкти, роблячи роботу зі сценою більш гнучкою.

Різноманітність інструментів і способів по створенню того чи іншого ефекту дозволяє підбирати більш зручний і ефективний варіант. У програмі є всі необхідні інструменти для створення флюїдних симуляцій, руйнувань і динаміки, роботи з рідинами і частинками.

### 4.1 Створення симуляції віртуального ефекту вибуху

Створюємо модель полігональної сфери та за допомогою контексту Geometry підключаємо ноду Mounting для того, щоб сфера перетворилась на потрібну нам форму (рис. 4.1).

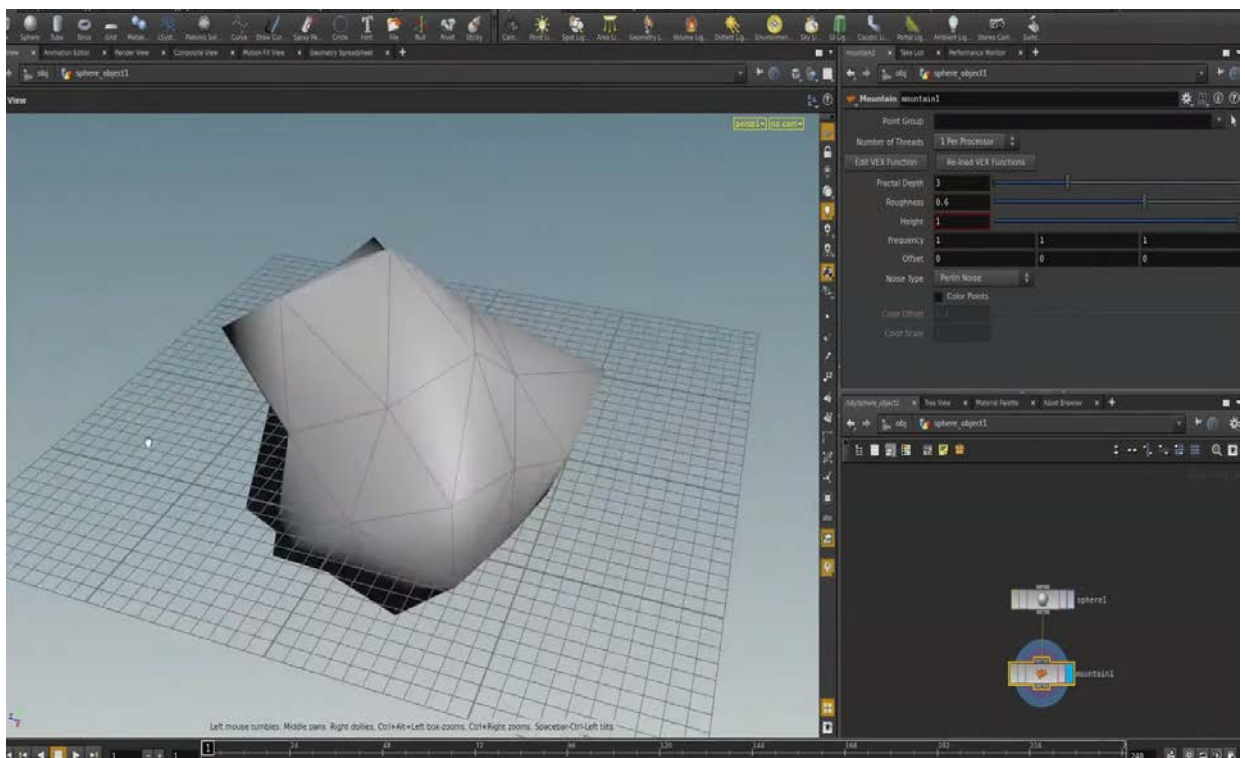


Рисунок 4.1 – Підключення ноди Mounting до полігональної сфери

У системі PyroFX настикаємо вкладку Explosion, та отримуємо стандартний скрип від програми Houdini, що змінює вид нашої полігонної сфери на туман світло-сірого кольору (рис. 4.2).

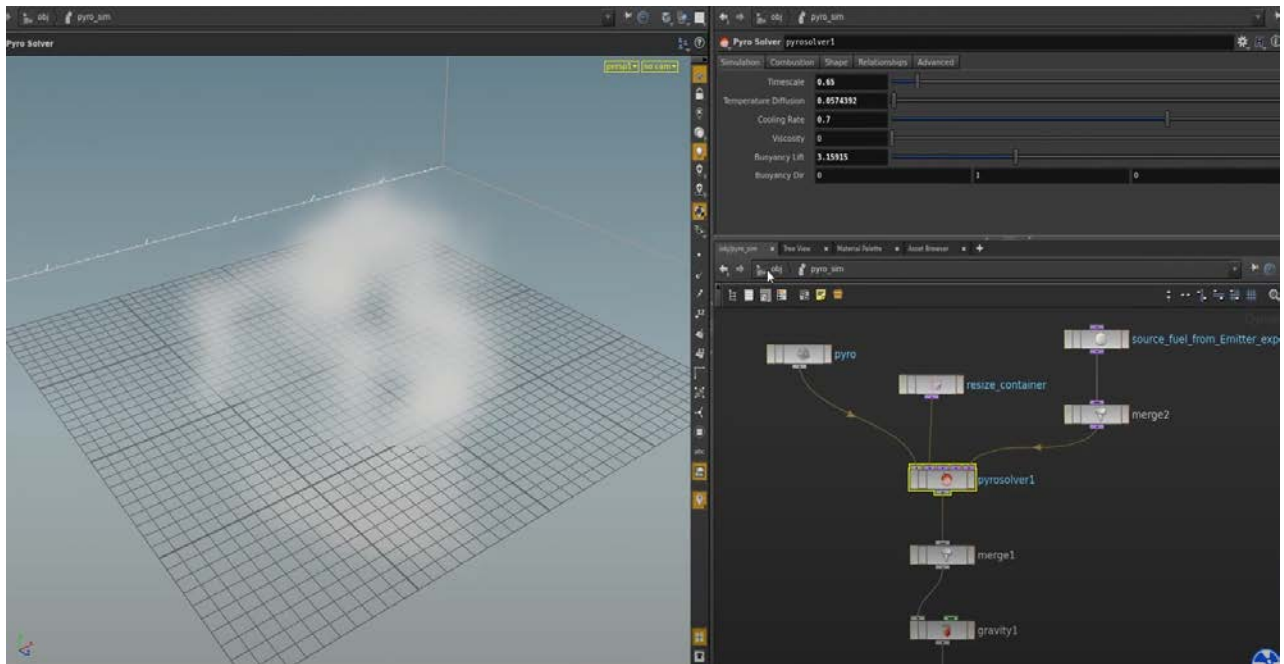


Рисунок 4.2 – Результат дії Explosion

Система Explosion нам автоматично видала дерево нод, з яких найголовнішими для налаштування нашого вибуху будуть:

1. create\_fuel\_volume - перетворює геометрію в пальне для Pyro FX.
2. pyro – відповідає за параметри щільності контейнера, тобто кількості пікселів, налаштування площин.
3. pyrosolver1 – відповідає за параметри симуляції динаміки.
4. resize\_container – відповідає за змінення розміру контейнера.
5. import\_pyrofields – імпорт всіх полів динаміки.
6. import\_pyro\_visualization – імпорт всіх полів з візуалізацією.

Наступним кроком буде налаштування розмірів контейнера вибуху за допомогою ноди Pyro задаємо потрібні нам значення щільності та розмірів (рис. 4.3).



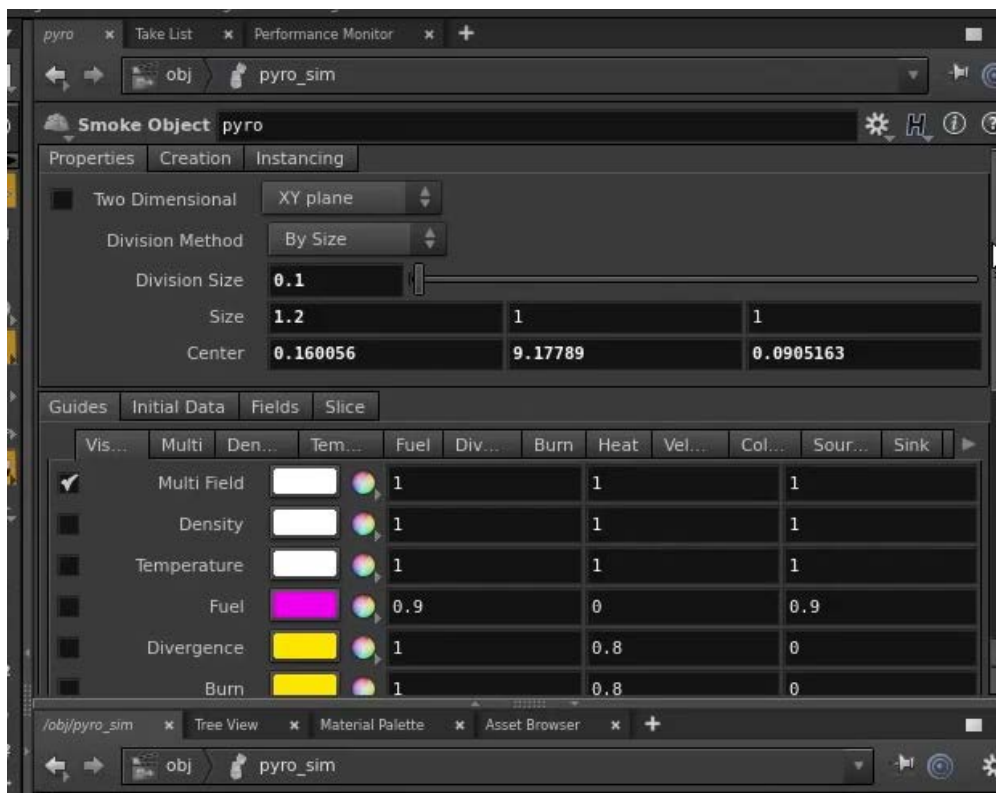


Рисунок 4.3 – Задані розміри за нодою Pyro

Далі за допомогою клавіші Tab ми додаємо у схему нод ще одну ноду, яка називається Gas\_Vortex\_Confinement. Ця нода за допомогою налаштувань робить наш вибух більш детально пророблений та реалістичний. Результат налаштування проработки можна побачити на рис. 4.4.

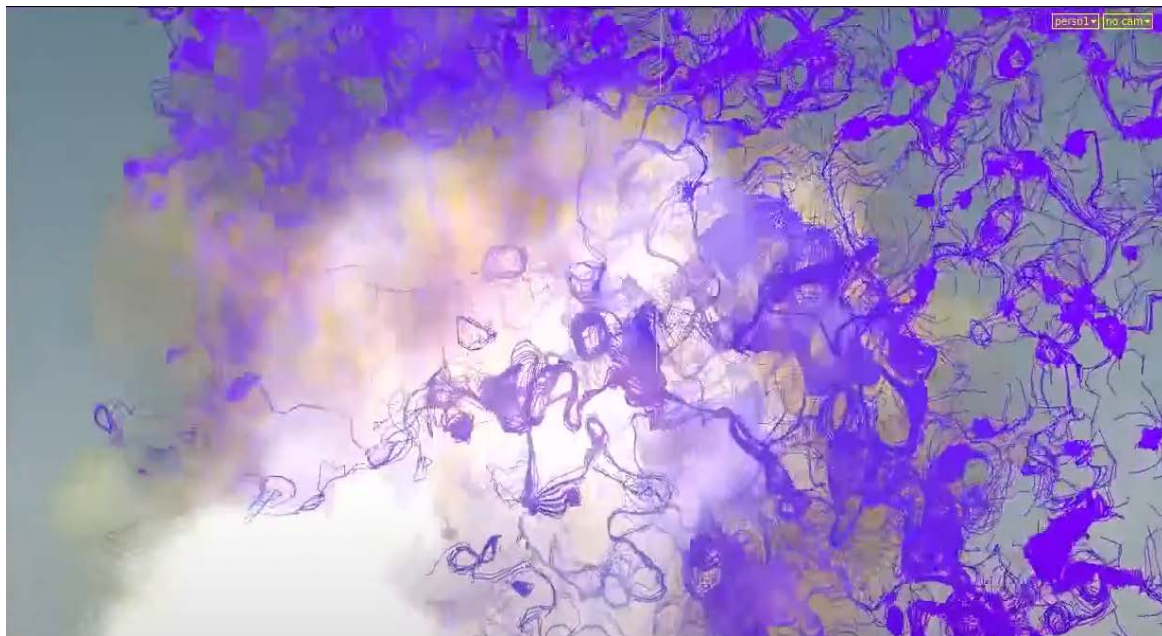


Рисунок 4.4 – Візуалізація ноди Gas\_Vortex\_Confinement

Ноду Gas\_Vortex\_Confinement підключаємо до нашого вузла pyrosolve1 та збільшуємо значення Confinement Scale до 4. Далі у ноді pyrosolve1 треба змінити значення температури дифузії до 0.08 та значення Buoyancy Lift також змінюємо на 4, інші налаштування залишимо. У вкладці Shape значення Disturbance підніmemo до 1.25 та Shredding підніmemo до 1.3. У вкладці Turbulence збільшимо показник Swirl Size до 0.85 та показник Turbulence до 3. У ноді Pyro ставимо значення Division size – 0.05. На цьому буде завершено налаштування симуляції.

Наступним кроком буде повернення до контексту сцени та за допомогою клавіши Reset Simulation будет виконано сброс симуляції. Далі потрібно зберегти на диск кеш симуляції за допомогою ноди import\_pyrofields (рис.4.5).

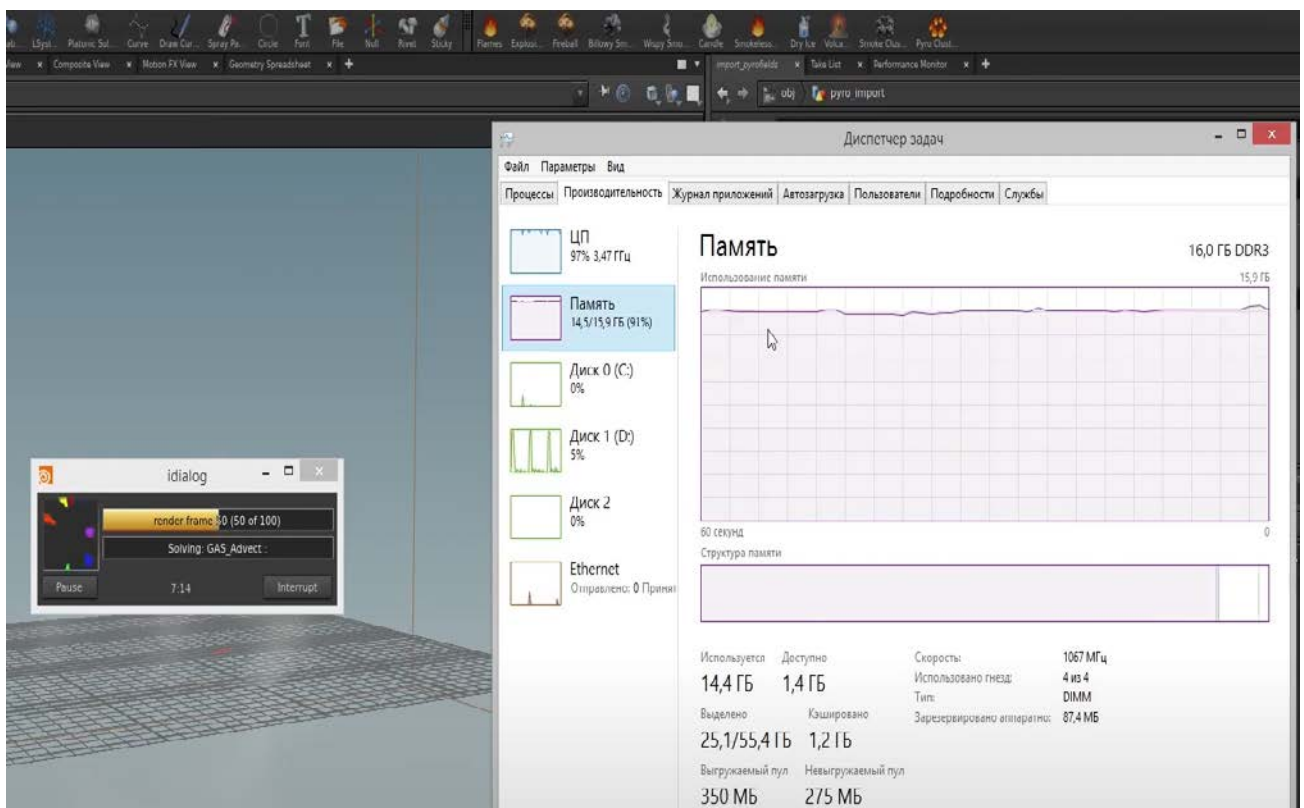


Рисунок 4.5 – Імпорт симуляції нодою import\_pyrofields

Вигляд отриманої симуляції вибуху після проведення рендерінгу за нодою import\_pyrofields можна побачити на рис.4.6.

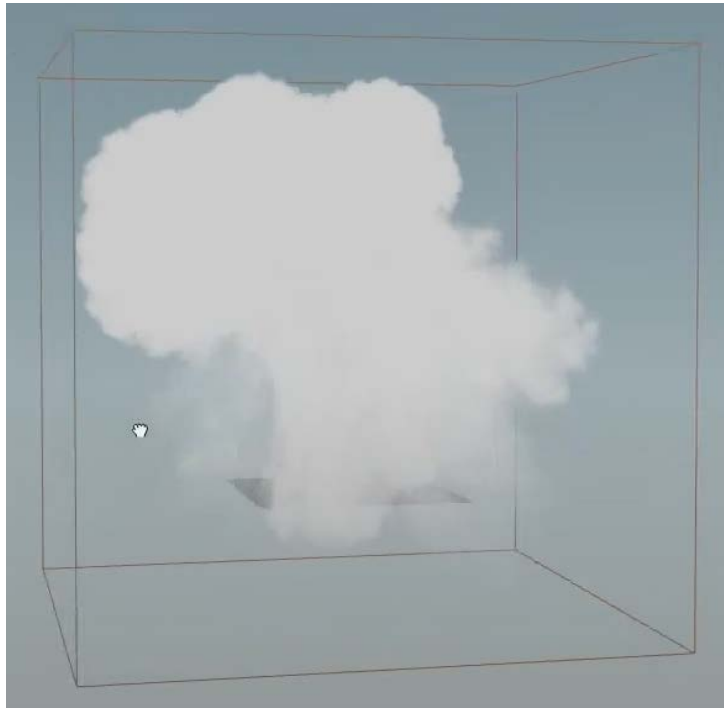


Рисунок 4.6 – Результати рендерингу симуляції вибуху

Для того, щоб додати світлового ефекту вибуху потрібно з зажатою клавішею Cntr натиснути на вкладку Spot Lighting та у цій системі нод натиснути на ноду spotlight1 і у налаштуваннях кольору обрати потрібний відтінок кольору для найбільш реалістичнішого вибуху (рис. 4.7).

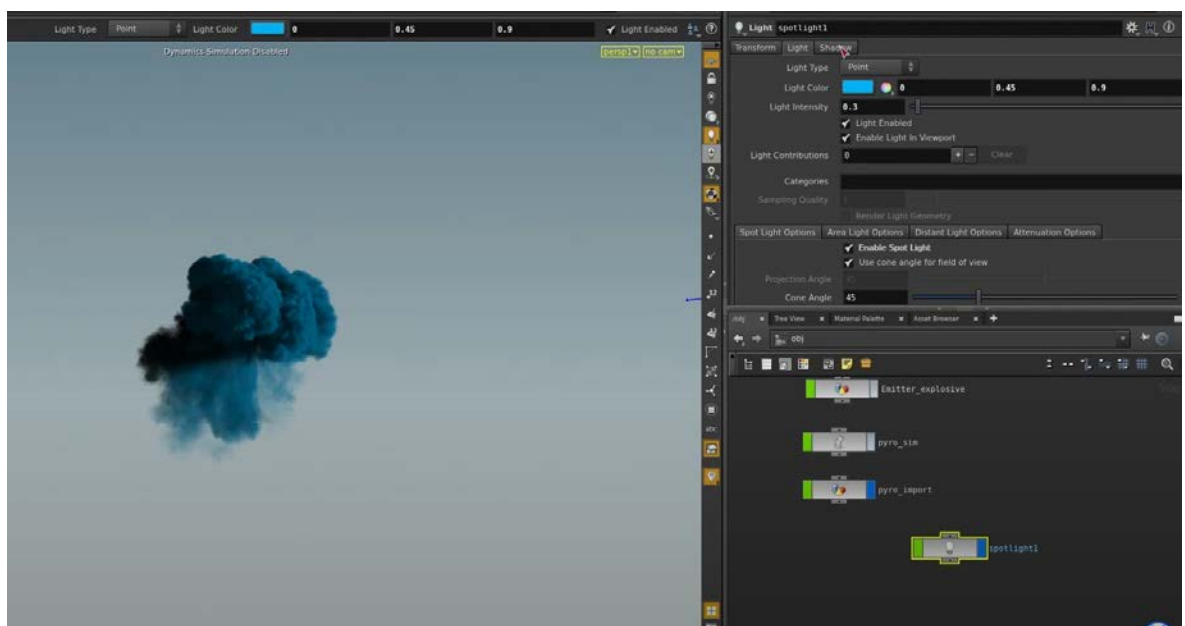


Рисунок 4.7 – Результат зміни кольору нодою spotlight1

Далі потрібно додати ще один відтінок кольору, для цього треба знов зажати клавішу Cntr та натиснути на вкладку Spot Lighting та у цій системі нод

натиснути на ноду spotlight2 і у налаштуваннях змінити інтенсивність на 0.7 (рис.4.8).

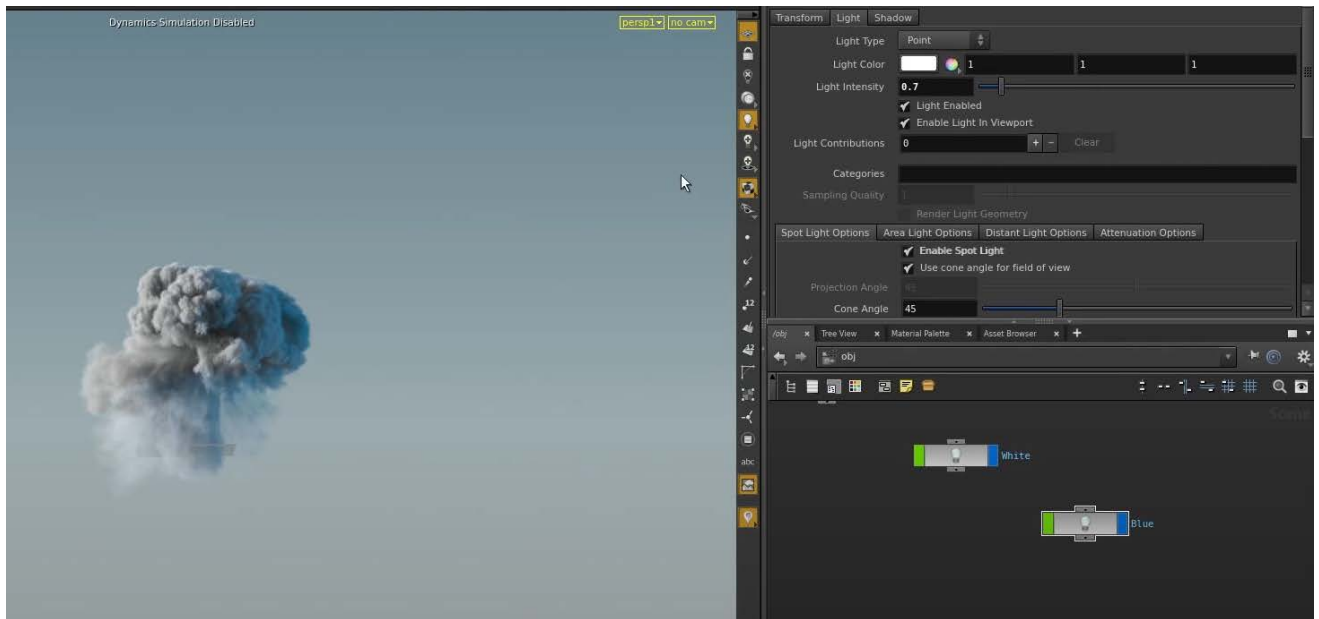


Рисунок 4.8 – Результат зміни кольору нодою spotlight2

Щоб створити камеру треба з зажатою кнопкою Cntr натиснути лівою кнопкою миші на вкладку с камерою. В налаштуваннях камери у вкладці View у параметрі Resolution задаємо границі кадру 720 на 1000 (рис. 4.9). Та натиснути на замочок, для того, щоб приблизити вид вибуху на камеру.

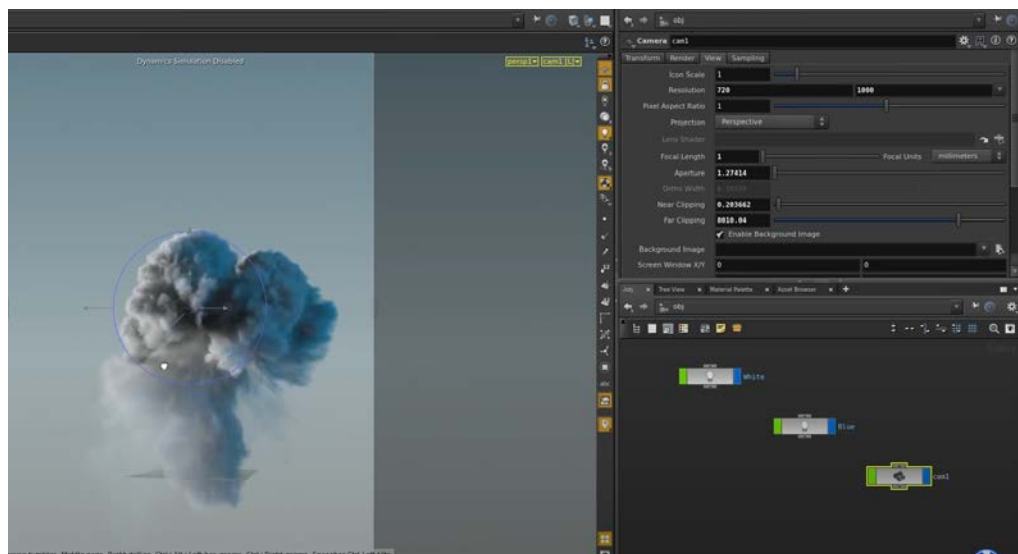


Рисунок 4.9 – Приближення вибуху за допомогою камери

## 4.2 Налаштування шейдера fireball та Render

Для того щоб перейти до шейдера, треба натиснути на контекст shop та у налаштуваннях стандартного шейдера fireball перейти у вкладку Fire. Тут ми бачимо ще дві вкладки: Density та Color. У налаштуваннях Density збільшуємо амплітуду до 15 (рис. 4.10).

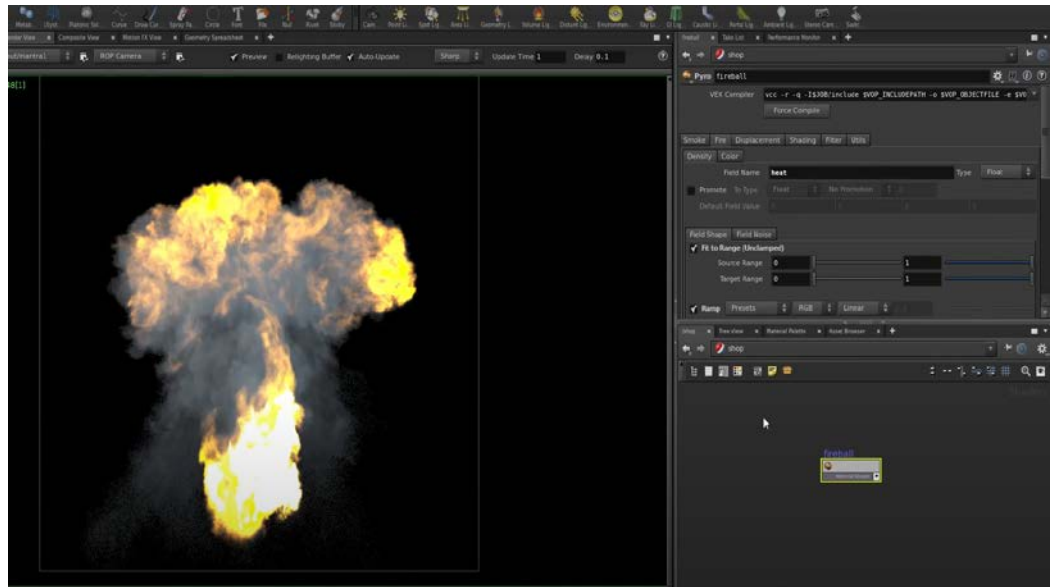


Рисунок 4.10 – Збільшення амплітуди вогню у Density

Переходимо у налаштування Color та підбираємо найбільш реалістичний колір для вогню вибуху за градієнтом (рис. 4.11).

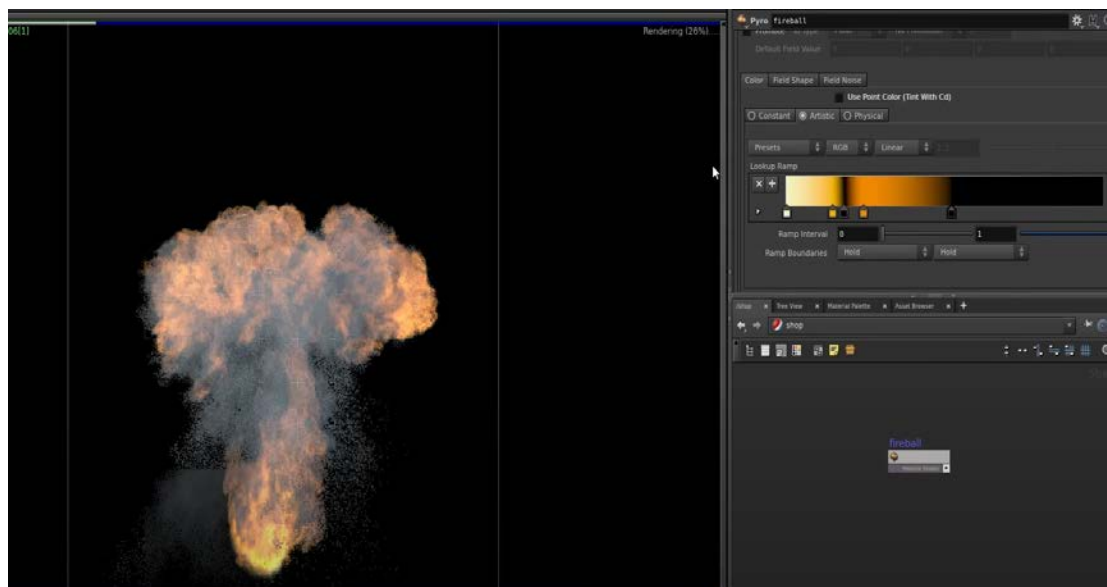


Рисунок 4.11 – Зміна кольору вогню за градієнтом



Після налаштувань вогню переходимо до диму у вкладці Smoke та віключаємо градієнт у Density та збільшуємо значення амплітуди до 10 (рис. 4.12).

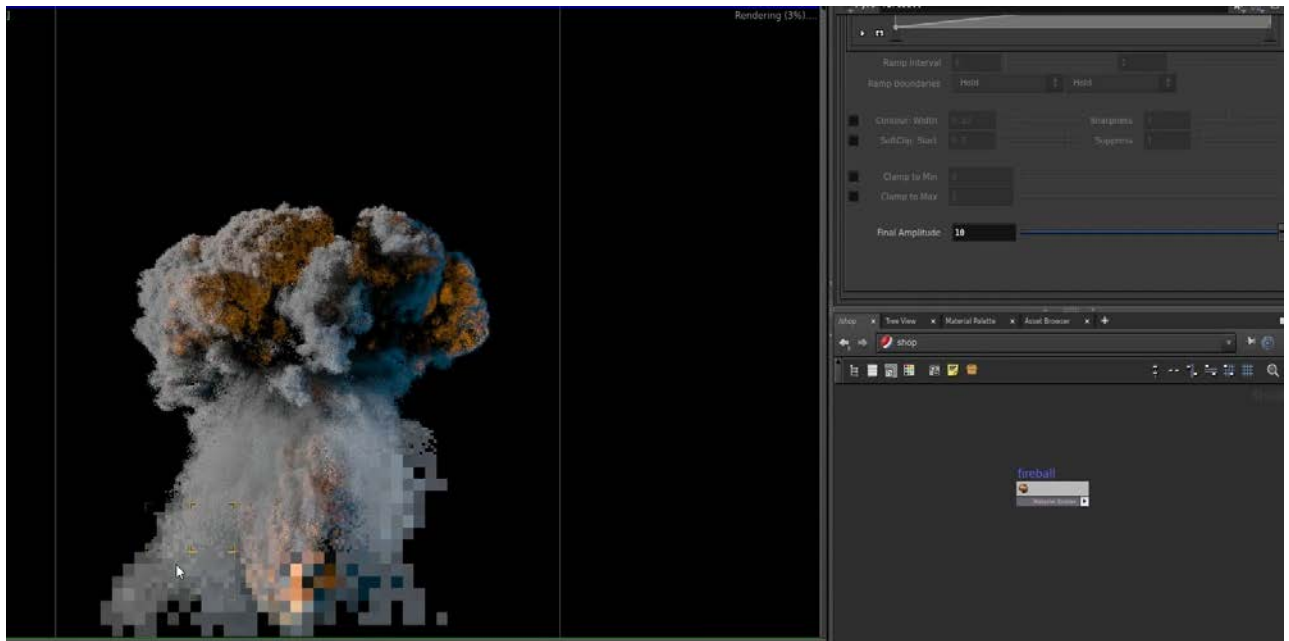


Рисунок 4.12 – Результат збільшення амплітуди диму

Переходимо у налаштування Color та підбираємо градієнт колір для диму. Для того, щоб зробити дим ще більш чорнішим зайдом у у вкладку Edit та виберем Color settings. У вкладці Color Settings забираємо галочку з COP and Render Viewports та натискаємо Apply (рис. 4.13).

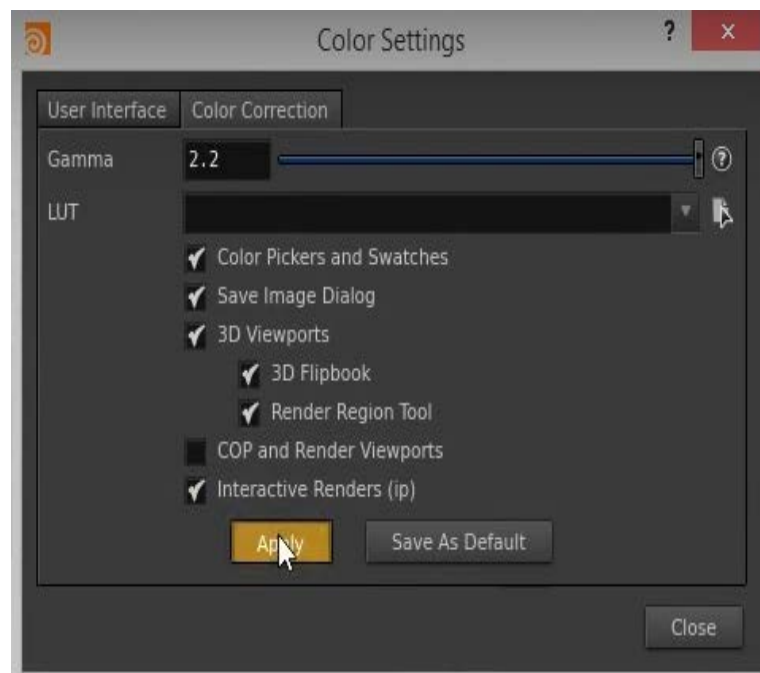


Рисунок 4.13 – Налаштування Color Settings

Для того, щоб додати більше внутрішнього світла у вибуху для більшої реалістичності треба повернутися до Scene View та у контексті сцени виділяємо ноду `pyro_import` та застосуємо до неї `Volume_light`. У ноді `volumelight1` знижуємо інтенсивність до 0.1 та відключимо вкладку `Shadow`.

Далі додаємо градієнт на фон вибуху, загрузивши фото у ноду `cam1` у строчці `Background Image` (рис. 4.14).

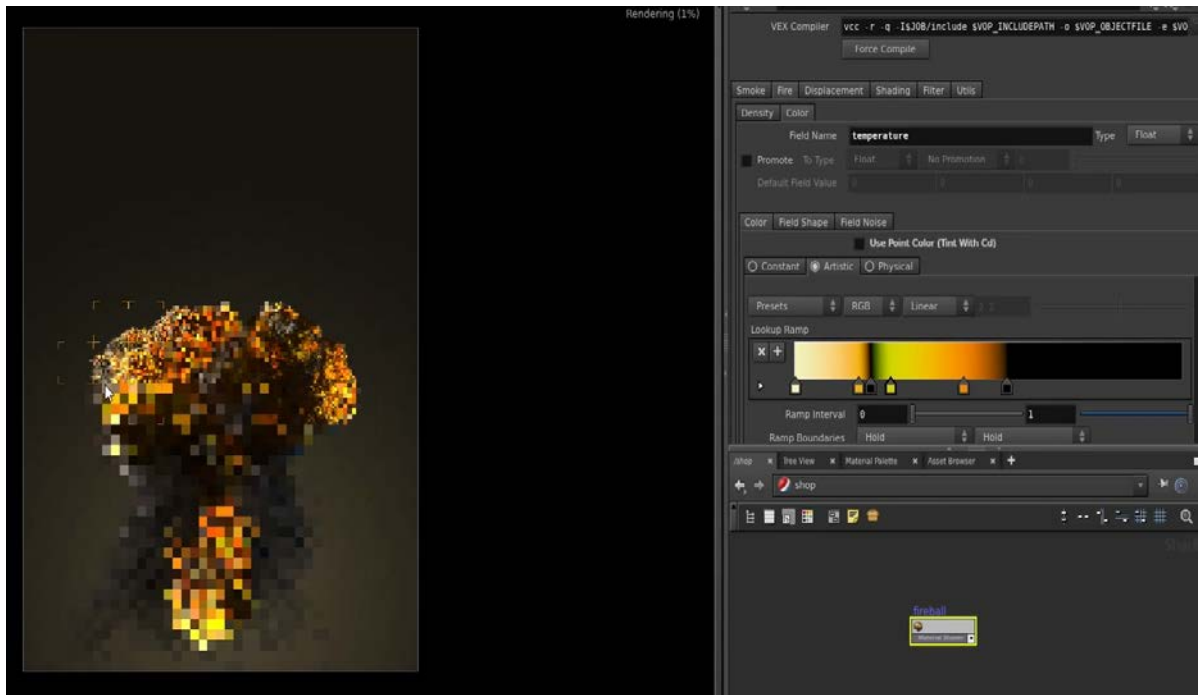


Рисунок 4.14 – Вибух з фото градієнтом на фоні.

Підключаємо ноду `Mantra` та робимо налаштування рендерінгу, які можна побачити на рис. 4.15 та запускаємо `Render`. Результат кінцевого рендерингу вибуху можна побачити на рис. 4.16.

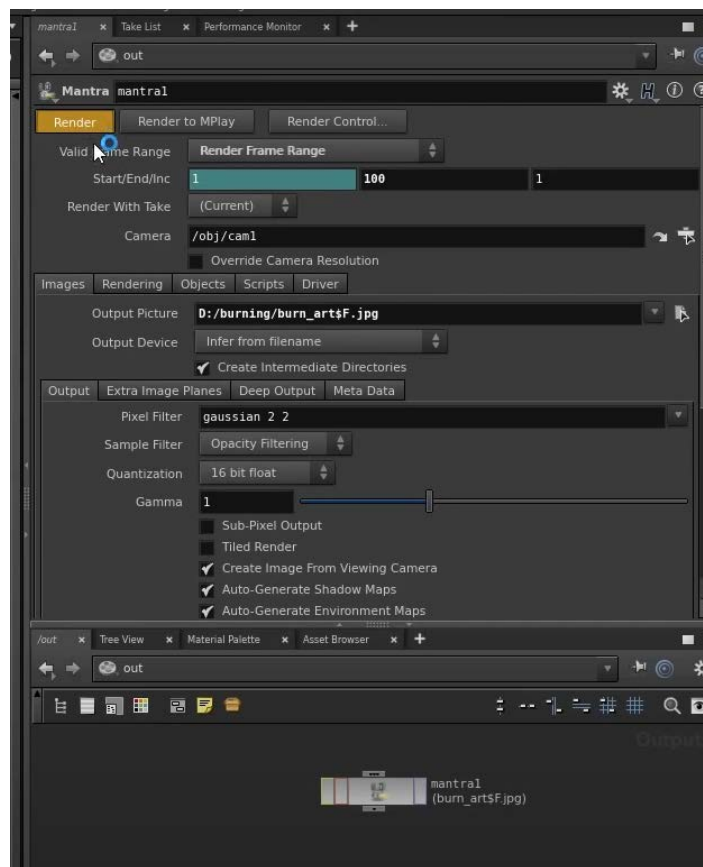


Рисунок 4.15 – Налаштування та запуск Render



Рисунок 4.16 – Ефект вибуху після рендерингу



## Висновки по розподілу 4

У четвертому розділі я розробив алгоритм створення ефекту вибуху в програмі Houdini. Розробив створення симуляції віртуального ефекту вибуху у системі Pyro FX. Проаналізував роботу та налаштування нод Gas\_Vortex\_Confinement, import\_pyrofields , spotlight1, роботу з нодою показу камери вибуху та додавання шейдера fireball та підбір градієнту кольору для найбільш реалістичного вигляду віртуального ефекту вибуху.

## ВИСНОВКИ

Комп'ютерні технології у сучасному кінематографі розкривають безліч нових можливостей перед режисерами, операторами, постановниками.

У цій дипломній роботі я провів дослідження програмних особливостей різних режимів налаштування цифрового композитингу в середовищі Nuke та розробив алгоритм створення віртуального ефекту вибуху в програмі Houdini. Зокрема, в рамках дослідження:

У основній частині дослідження розглядалися основні підходи та алгоритми роботи з віртуальними ефектами та композитинг в NukeX, за якими доцільно проводити налаштування Nuke для 3D сцени. Сформулював основні особливості переміщення камери в 3D сцені NukeX. Розробив алгоритм переміщення віртуальних 3D-об'єктів в часі. Проаналізував основні функції та можливості нодів, які відповідають за рух в сцені, суміщення 3D об'єктів в просторі. В окремій частині дослідження створив алгоритм створення віртуального ефекту вибуху, проаналізував основні функції та можливості нодів, які відповідають за інтенсивність, амплітуду, колір та реалістичність віртуального ефекту вибуху.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Арнхейм Р. Кино как искусство. М. – изд. Прогресс, 2011. С. 169.
2. Трудности композитинга при работе в 3D-стерео. URL: <http://www.mediavision-mag.ru/> (дата звернення: 26.10.2018).
3. Goh S. Digital Compositing with Nuke 101. Hong Kong, 2010. 160 p., ISBN: 978-988-18465-1-8.
5. Okun A., Susan Z. Visual Effects Society. China, 2010. 923 p., ISBN 978-0-240-81242-7
4. Ganbar R. Nuke 101. Professional Compositing and Visual Effects. USA, 2011. 404 p., ISBN: 978-0-321-73347-4
6. Райт С. Цифровой композитинг в кино и видео / Райт Стив; пер. с англ. Медведникова М.М. – изд. 2-е, доп. и перераб. – М. : НТ Пресс, 2009. – 448 с. : ил. – (Секреты профессионалов). ISBN 978-5-477-00621-2.
7. John G., [digital] VISUAL EFFECTS & COMPOSITING. USA, 2015. 423 p., ISBN 13: 978-0-321-98438-8.
8. B. Bratt, Rotoscoping. Techniques and Tools for the Aspiring artist., London: Elsevier, 2011, p. 283. ISBN: 978-0240817040.
9. S. Wright, Digital Compositing for Film and Video., Burlington: Focal Press, 2013, p. 475. ISBN-13: 978-0240813097.
10. Орлов Б. В. Дослідження програмних особливостей створення 3d елементів для сучасного цифрового кіно Elconf-2018. – 2018.- Секція №7. – С 359 – 362.

## THE SUMMARY

Virtual effects and special effects - a technological technique in cinema, television, shows and computer games, used to visualize scenes that can not be shot in the usual way (for example, as to visualize the scene of battleships or spacecraft in the distant future).

Special effects are also often used when natural scene shooting is too costly compared to a special effect (for example, shooting a large-scale explosion). Virtual effects are also used to improve or modify previously filmed videos (for example, to superimpose a weather map on the background for a TV presenter on live TV channels, which tells about the weather forecast).

The digital image changes our sense of the necessary connection between the camera and reality. The presence of both is no longer absolutely necessary. Now it is much easier to "photograph" what is impossible to see. Computer technology and effects turn images into pixels that can be easily transformed, processed, modified. The line between animation (which creates images where they did not exist before) and editing (which deals with the permutation of fragments of events that occurred in front of the camera) is erased. When the artist is able to easily manipulate the digital image, either as a whole or frame by frame, the film becomes a series of drawings. The ability to manually draw on digital images is a very serious change in the status of cinema, paradoxically returning the "art of moving images" to its origins.

Special effects were born almost at the same time as cinema. Virtual effects are also often used when too much money is spent on the natural shooting of a scene compared to a special effect (for example, if it is a very large-scale explosion of a structure). Special effects are also used to improve or modify previously filmed videos (for example, inserting a background image of a TV presenter who talks about the weather forecast).

Freeze frame. Used in scenes where you want to show the sudden appearance or disappearance of an object or change the natural interval between two events.

Influx. Cinema of the early and mid-twentieth century used this technique to show the "transformation" of one object into another. On the same segment of the film on the same background is shot first one object with a gradual decrease in exposure (closing the aperture of the lens or - more often - a decrease in the angle of the shutter), and then - another with a gradual increase in exposure. As a result, the image of one object on the screen is smoothly replaced by another.

Frame-by-frame shooting. The method is used mainly when working with layouts; widely used in the middle of the XX century in making films about monsters (for example, "King Kong"). The same technology is used to make puppet and plasticine cartoons (animations). Frame-by-frame shooting allows you to make visible slow processes, invisible to the eye: plant growth, diurnal movement of lights, etc.

Accelerated filming or "repid". High-speed shooting of a process that proceeds very quickly (for example, an explosion). Then the film scrolls at a normal pace, which allows you to get a slow-motion high-quality image.

Slow motion, or "accelerated playback." The effect is the opposite of rapid - shooting is done at a lower speed, and then scrolls at a normal or accelerated pace. Slight acceleration is used in fight scenes, when you need to remove the rapid movements that the actors simply would not have time to perform. Strong acceleration can be used to create a comic effect or to display large periods of time.

Reverse shooting. For example, when you want to remove a person taking off, it is easier to remove the fall, and then scroll the film in the opposite direction.

Special effects are divided into visual - these are optical effects and computer graphics, mechanical and sound special effects.

The most powerful feature in Nuke is its 3D module. Nuke has almost the entire multifunctional 3D system, which can import all cameras, make simple objects, edit and import objects, make camera projections, extract various types of 2D data from 3D content and much more. While the capabilities of creating virtual effects in Nuke are great for the compositing system, they still don't fully do the job of replacing full-fledged 3D packages. Also, the Nuke 3D module is used to make it easier to perform

2D tracking of information, simplify rotoscoping and much more. Nuke also works well with external 3D applications and can communicate with them using file formats commonly used in 3D applications: Alembic, OBJ and FBX. .Obj extension for geometry-only files. Alembic and FBX file types can contain almost anything generated in a 3D scene.

The world position pass is a pass that is rendered in 3D software that does not display the visual part of the image, but it has certain advantages. The world position bar displays where each pixel of the image is located according to the zero coordinate (0,0,0). Using this pass and the PositionToPoints node you can create a cloud of pixels that will be located at their correct positions in the 3D world.

To import the camera from third-party sources, you must perform the following algorithm. When performing 3D compositing, it is necessary to have a camera that would behave similarly to the one that filmed this scene. First of all, you need to set up a 3D scene.

Nuke 3D capabilities are designed to make it easier for the compositing department to connect with the 3D department. In addition, if there are 3D objects in the scene, the composer can obtain 3D tracking data for use for their own purposes. Also, the 3D Nuke module is used to make it easier to perform 2D tracking information, simplify rotoscoping and much, much more. Nuke works well with external 3D applications and can communicate with them using file formats commonly used in 3D applications: Alembic, OBJ and FBX. .Obj extension for geometry-only files. Alembic and FBX file types can contain almost anything generated in a 3D scene. In this thesis, I studied the software features of different modes of setting up digital compositing in the Nuke environment and developed an algorithm to create a virtual explosion effect in Houdini.

Quite often, when creating effects for movies, commercials, various videos, use the program Houdini. The components of this program allow you to effectively connect different elements and objects, making work with the scene more flexible.

A variety of tools and ways to create a particular effect allows you to choose a more convenient and effective option. The program has all the necessary tools to create fluid simulations, destruction and dynamics, work with liquids and particles.

The main part of the study considered the main approaches and algorithms for working with virtual effects and compositing in NukeX, according to which it is advisable to configure Nuke for a 3D scene. Formulated the main features of moving the camera in the 3D scene NukeX. Developed an algorithm for moving virtual 3D objects in time. Analyzed the main functions and capabilities of nodes that are responsible for the movement in the scene, the combination of 3D objects in space. In a separate part of the study he created an algorithm for creating a virtual explosion effect, analyzed the main functions and capabilities of nodes that are responsible for the intensity, amplitude, color and realism of the virtual explosion effect.